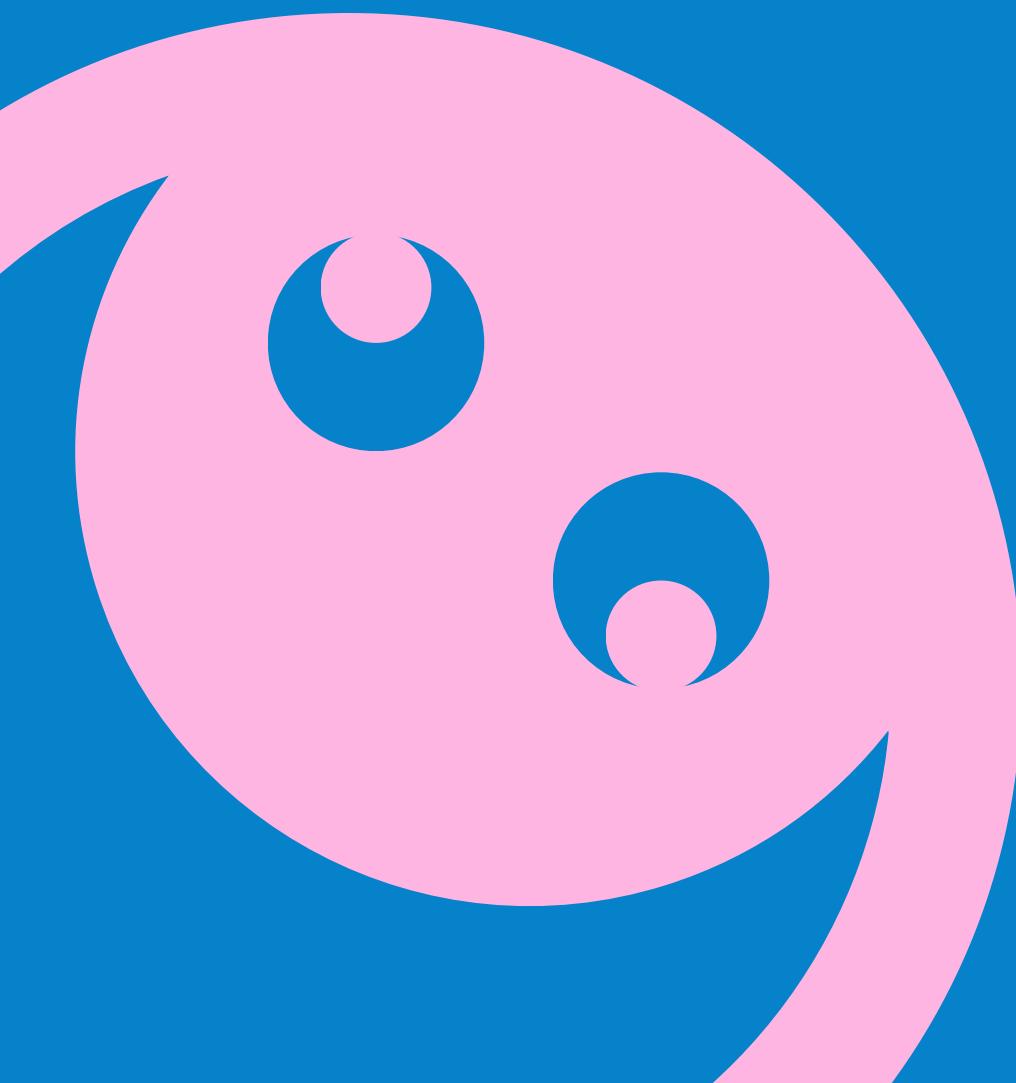


The University of Sydney
School of Aerospace, Mechanical, & Mechatronics Engineering
AERO4701 Space Engineering 3

Lunar Atmospheric Investigations with Cube-Sats
(LUNATICS)

ASSEMBLY, INTEGRATION, & TESTING REPORT



Kelly Chen
ADCS Lead

Austin Cleary
Structural Lead

Ellie Deveson
Payload Lead

Joshua Dickford
EPS Lead

James Hocking
Communications Lead

Jasmine Khuu
Thermal Lead

Aum Mehta
Systems Lead

William Ridley-Smith
OBC Lead

Contents

List of Figures	iv
List of Tables	v
1 Scope	1
1.1 Document Scope	1
1.2 Mission Overview	1
2 Testing Overview	2
2.1 Assembly Requirements	2
2.2 Initialisation Requirements	2
3 Mechanical	3
3.1 Test Definition	3
3.1.1 Requirements	3
3.1.2 Test Facilities, Setup and Configuration	3
3.1.3 Loads Adopted	4
3.1.4 Test Flow	4
3.2 Test Results	4
3.2.1 T-MEC-01	5
3.2.2 T-MEC-02	5
3.2.3 T-MEC-03	5
3.2.4 T-MEC-04	6
3.2.5 T-MEC-05	6
4 Thermal	7
4.1 Test Definition	7
4.1.1 Requirements	7
4.1.2 Test Facilities, Setup and Configuration	8
4.1.3 Loads Adopted	8
4.1.4 Test Flow	9
4.2 Test Results	9
4.2.1 T-THE-01	9
4.2.2 T-THE-02	11
5 OBC	12
5.1 Test Definition	12
5.1.1 Requirements	12
5.1.2 Test Facilities, Setup and Configuration	13
5.1.3 Loads Adopted	15
5.1.4 Test Flow	15
5.2 Test Results	16
5.2.1 T-OBC-01	16
5.2.2 T-OBC-02	16
5.2.3 T-OBC-03	17
5.2.4 T-OBC-04	17
5.2.5 T-OBC-05	17
5.2.6 T-OBC-06	17
5.2.7 T-OBC-07	17
5.2.8 T-OBC-08	17
5.2.9 T-OBC-09	17
5.2.10 T-OBC-10	18
6 EPS	19
6.1 Test Definition	19
6.1.1 Requirements	19
6.1.2 Test Facilities, Setup and Configuration	20
6.1.3 Loads Adopted	20
6.1.4 Test Flow	20



6.2	Test Results	21
6.2.1	T-EPS-01	22
6.2.2	T-EPS-02	22
6.2.3	T-EPS-03	22
6.2.4	T-EPS-04	22
6.2.5	T-EPS-05	22
6.2.6	T-EPS-06	23
6.2.7	T-EPS-07	23
6.2.8	T-EPS-08	23
6.2.9	T-EPS-09	23
7	Communications	24
7.1	Test Definition	24
7.1.1	Requirements	24
7.1.2	Test Facilities, Setup and Configuration	25
7.1.3	Loads Adopted	25
7.1.4	Test Flow	25
7.2	Test Results	26
7.2.1	T-TTC-01	26
7.2.2	T-TTC-02	26
7.2.3	T-TTC-03	26
7.2.4	T-TTC-04	26
7.2.5	T-TTC-05	26
7.2.6	T-TTC-06	27
7.2.7	T-TTC-07	27
8	ADCS	28
8.1	Test Definition	28
8.1.1	Requirements	28
8.1.2	Test Facilities, Setup and Configuration	29
8.1.3	Loads Adopted	31
8.1.4	Test Flow	32
8.2	Test Results	33
8.2.1	T-AOC-01	33
8.2.2	T-AOC-02	34
8.2.3	T-AOC-03	34
8.2.4	T-AOC-04	35
8.2.5	T-AOC-05	37
8.2.6	T-AOC-06	39
8.2.7	T-AOC-07	40
8.2.8	T-AOC-08	43
8.2.9	T-AOC-09	44
8.3	On-day Testing	44
9	Structure	45
9.1	Test Definition	45
9.1.1	Requirements	45
9.1.2	Test Facilities, Setup and Configuration	45
9.1.3	Loads Adopted	45
9.1.4	Test Flow	46
9.2	Test Results	46
9.2.1	T-STR-01	46
9.2.2	T-STR-02	46
9.2.3	T-STR-03	46
9.2.4	T-STR-04	47
9.2.5	T-STR-05	47
9.2.6	T-STR-06	47
10	Payload	48
10.1	Test Definition	48



10.1.1 Requirements	48
10.1.2 Test Facilities, Setup and Configuration	48
10.1.3 Loads Adopted	48
10.1.4 Test Flow	49
10.2 Test Results	49
10.2.1 T-PAY-01	49
10.2.2 T-PAY-02	50
10.2.3 T-PAY-03	50
10.2.4 T-PAY-04	50
References	52
A Mission requirements	53
A.1 CubeSat System Requirements	53
A.2 Payload Requirements	59
B Code	60
B.1 OBC Code	60
B.2 ADCS Code	60
B.3 Ground Station Code	60
C AIT Video	60



List of Figures

1	LUNATICS-0 System Layout	1
2	Temperatures at the central point of each external face are shown for the most extreme conditions over four orbital cycles [1]	8
3	Temperature distribution at the peak of the cyclic thermal test	10
4	Temperature distribution at the trough of the cyclic thermal test	10
5	Temperature distribution throughout the cyclic thermal test	10
6	Temperature distribution at the end of the thermal test	11
7	Temperature distribution throughout the bake-out thermal test	11
8	Testing setup for the OBC	13
9	Frontend display that has the ability to override the states on LUNATICS-0. Delete any SU data, and to read the current time on the OBC. Labels have been added to show what requirements have been met	14
10	SQLite Database holding downlinked payload data	14
11	Downlinked WOD data. Timestamp shows data being downlinked every 30s	16
12	UTC Time from the internet at time of recording data in Figure 11	16
13	EPS testing setup with one cell to be connected	20
14	PCB Stack EPS Testing	21
15	INA219 current sensor package pinouts	22
16	GUI tabs for display of data	27
17	Magnetorquer testing setup	30
18	Picture of T-AOC-05 with the IMU Connected to the Raspberry Pi Pico on a BreadBoard	31
19	Sun sensor testing with phone light passover	34
20	Current measured across the Y-Axis Magnetorquer using ADCS PCB	35
21	Current measured across the Z-Axis Magnetorquer using ADCS PCB	35
22	X-Axis Magnetorquer across duty cycle	36
23	Y-Axis Magnetorquer across duty cycle	36
24	Z-Axis Magnetorquer across duty cycle	36
25	Euler Angles Obtained from BNO085 IMU Test	37
26	Quaternions Obtained from BNO085 IMU Test	37
27	Calibrated Acceleration Obtained from BNO085 IMU Test	38
28	Calibrated Gyroscope Obtained from BNO085 IMU Test	38
29	Magnetic Field Strength Obtained from BNO085 IMU Test	39
30	Estimated Quaternion Using Filler Sensor Measurements	39
31	Estimated Euler Angles Using Filler Sensor Measurements	40
32	Estimated Angular Velocity Using Filler Sensor Measurements	40
33	Estimated Quaternions with Sensor Values When IMU is Sitting Idle	41
34	Estimated Euler Angles with Sensor Values When IMU is Sitting Idle	41
35	Estimated Angular Velocity with Sensor Values When IMU is Sitting Idle	42
36	Estimated Quaternions with Sensor Values When IMU is Conducting a 360° Roll	42
37	Estimated Euler Angles with Sensor Values When IMU is Conducting a 360° Roll	43
38	Estimated Angular Velocity with Sensor Values When IMU is Conducting a 360° Roll	43
39	Testing of attitude control algorithm software	44
40	Anodised components of satellite (with pink shade)	47
41	Payload testing showing red LED (power) on	49
42	Unique emission spectra for different coloured filters (atmospheric compositions)	50
43	Results from live demonstration of payload	50
44	GUI depending on whether satellite is in or not in PAYLOAD state	51



List of Tables

1	Mechanical Testing	3
2	Mechanical Results (FEA and Preliminary Physical Test)	4
3	Quasi-static acceleration FEA results	5
4	Modal Analysis FEA results	5
5	Sinusoidal vibration FEA results	5
6	Random vibration FEA results	6
7	Shock FEA results	6
8	Thermal Testing	7
9	Thermal Vacuum Cycling Requirements	8
10	Thermal Results	9
11	OBC Testing	12
12	OBC Results	16
13	EPS Testing	19
14	EPS Results	21
15	Communications Testing	24
16	Communications Results	26
17	ADCS Testing	28
18	ADCS Results	33
19	Maximum dipole moment produced by each of the magnetorquers.	34
20	Structures Testing	45
21	Structures Results	46
22	Payload Testing	48
23	Payload Results	49
24	Structural system requirements.	53
25	Attitude & Orbit Control Subsystem (AOCS) requirements.	54
26	Electrical Power System (EPS) requirements.	54
27	Onboard Computer & Data Handling (OBC/OBDH) requirements.	55
28	Telemetry, Tracking & Command (TT&C) requirements.	56
29	Thermal Control Subsystem Requirements	57
30	General system requirements.	58
31	Science and payload mission requirements.	59



1 Scope

1.1 Document Scope

This report presents an outline of the tests conducted on the LUNATICS-0 Cube-Sat. Each subsystem on the satellite underwent a set of pre-specified tests to ensure it could meet the outlined requirements for operation in orbit. The tests outlined in the Assembly, Integration and Testing Plan document were followed to ensure consistency. This allowed validation of the performance of each subsystem within the overall satellite context for optimal operation.

1.2 Mission Overview

Although lunar orbiters have studied the Moon for decades, its exosphere remains poorly understood. Unlike Earth's dense atmosphere, the Moon's exosphere is a sparse layer of particles with minimal interaction, yet it holds key insights into processes that have shaped the moon and other bodies with thin atmospheres both in our solar system and beyond.

Studying the composition and behaviour of exospheric gases such as Helium, Neon, and Argon can reveal their sources, for example solar wind, micrometeorites, or lunar outgassing and support future exploration through In-Situ Resource Utilisation (ISRU).

Our first satellite, LUNATICS-0, will operate in Low Earth Orbit (LEO) at 550 km to validate sensor technologies in preparation for lunar deployment. It will target gases common to both Earth's upper atmosphere and the lunar exosphere, including Helium and Argon, for calibration and performance testing.

Future missions aim to deploy a dedicated lunar orbiter equipped with enhanced sensors to map the spatial and temporal distribution of exospheric gases, contributing to lunar science, ISRU feasibility, and mission design—building on efforts like LADEE, Chandrayaan-2, and Artemis.

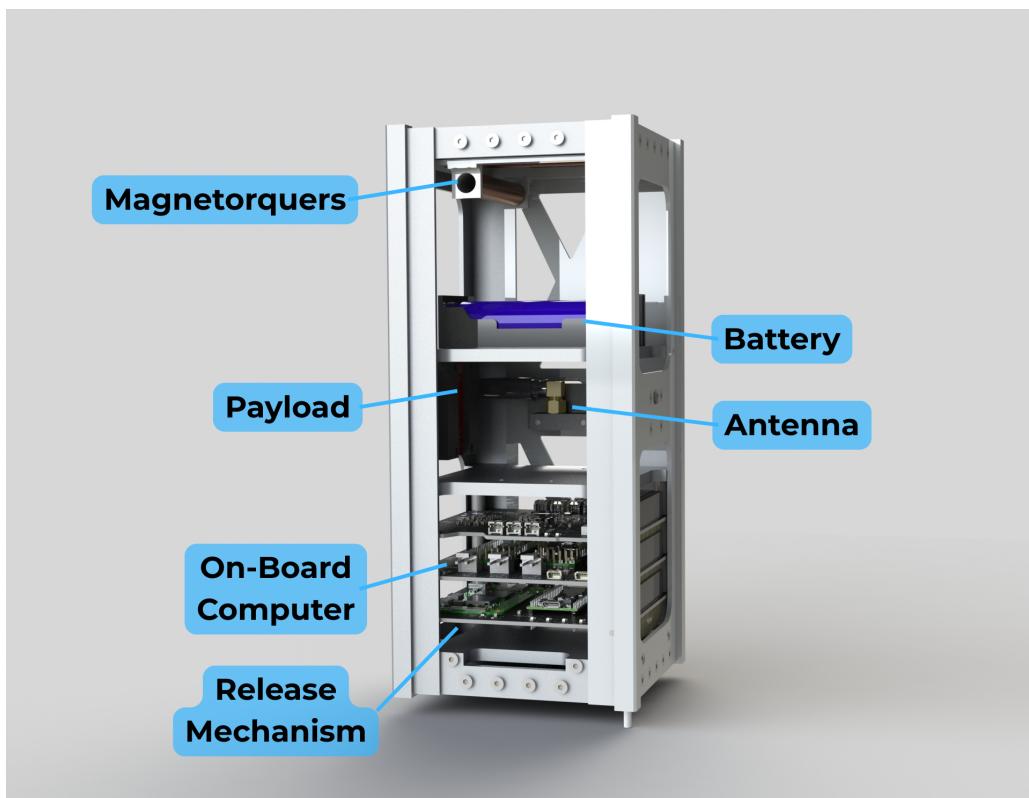


Figure 1: LUNATICS-0 System Layout.

2 Testing Overview

2.1 Assembly Requirements

Prior to satellite testing, the assembly needed to be verified to ensure it met the appropriate requirements. Launch requirements include the assembly mass being under 2.66 kg, the assembly being dimensionally accurate and containing all remove before flight tags. The LUNATICS-0 CubeSat met each of these launch requirements. To ensure assembly quality the satellite needed to not require any post-assembly modifications. This requirement was satisfied. The assembly quality was also tested using hand held vibration testing. LUNATICS-0 was able to withstand these vibrations and deemed to be of satisfactory quality.

2.2 Initialisation Requirements

Prior to launch, the CubeSat remains in a non-powered state. The Remove Before Flight pin physically inhibits any power flow from the batteries. Once removed, the system still remains dormant due to the deployment switches. As the CubeSat is ejected from the deployer, the deployment switches are released and will allow power to flow, which turns the On-Board Computer (OBC) on, activating the satellite in its initialisation state.



3 Mechanical

3.1 Test Definition

3.1.1 Requirements

Table 1: Mechanical Testing

Test ID	Req.	Description	Method	Pass Criteria
T-MEC-01	S-2.1.1	Quasi-static Acceleration	FEA simulation and lab test	The CubeSat shall withstand, without any degraded performance, the quasi-static acceleration indicated by S-2.1.1
T-MEC-02	S-2.2.1 S-2.2.2	Natural Frequency Survey	FEA simulation and lab test (one before vibrations and shock testing and one after)	Lowest natural frequency of the FM of the CubeSat shall be > 90 Hz
T-MEC-03	S-2.3.1	Sinusoidal vibrations	FEA simulation and lab test	The CubeSat shall withstand, without any degraded performance, the sinusoidal vibrations indicated by S-2.3.1
T-MEC-04	S-2.4.1	Random vibrations	FEA simulation and lab tests	The CubeSat shall withstand, without any degraded performance, the random vibrations indicated by S-2.4.1
T-MEC-05	S-2.5.1	Shock Loads	FEA simulation and lab tests of shocks along all axes twice	The CubeSat shall withstand, without any degraded performance, the shock levels indicated by S-2.5.1

The requirements for the mechanical subsystem can be found in the provided requirements document.

3.1.2 Test Facilities, Setup and Configuration

All mechanical tests conducted to date have been performed using Finite Element Analysis (FEA) in ANSYS Mechanical. The model includes the primary structural and mass components of the LUNATICS-0 CubeSat, and was developed to simulate quasi-static loads, vibrational modes, random vibrations, and shock loads in accordance with the relevant requirements.

Physical structural testing, including vibration, shock, and modal analysis using laboratory equipment, is future work that would improve readiness. These tests would be conducted at a certified structural test facility using equipment conforming to ISO/IEC 17025 standards. The CubeSat would be mounted on a rigid fixture simulating the launch vehicle interface, with accelerometers used to monitor structural response. Tests would include vibration analysis using an electrodynamic shaker table, shock testing using a pneumatic impact hammer and drop tower, and modal testing with a modal hammer and accelerometer array.



3.1.3 Loads Adopted

The mechanical loads assessed during testing were based on mission requirements and were evaluated through Finite Element Analysis (FEA). Only a basic physical shake test was conducted to check for component looseness. The adopted load conditions for simulation and qualitative testing are as follows:

- **Quasi-static Load:** 13 g applied independently along each principal axis, simulated in FEA.
- **Natural Frequency Test:** FEA simulations identified the six lowest natural frequencies of the CubeSat structure.
- **Sinusoidal Vibration:** Simulated using frequency sweeps from 5–100 Hz at 2.5 g and 100–120 Hz at 1.25 g amplitude.
- **Random Vibration:** Simulated using a Power Spectral Density (PSD) profile over 20–2000 Hz, resulting in an RMS acceleration of 8.03 g.
- **Shock Testing:** FEA simulated using a response spectrum of up to 4000 g and 10000Hz applied twice in all axes.

3.1.4 Test Flow

The current mechanical testing flow consists of numerical simulations complemented by limited physical testing. The planned and executed sequence is as follows:

1. FEA-based modal analysis to determine baseline natural frequencies.
2. FEA-based quasi-static acceleration analysis.
3. FEA-based sinusoidal vibration analysis.
4. FEA-based random vibration analysis.
5. FEA-based shock load simulation.
6. Physical shake test to detect any loose internal components.
7. Visual inspection post-shake test.

3.2 Test Results

Table 2: Mechanical Results (FEA and Preliminary Physical Test)

Test ID	Results	PASS/FAIL
T-MEC-01	FEA results confirmed the structure withstood 13 g static loads in all directions without exceeding yield criteria.	PASS
T-MEC-02	FEA modal analysis identified the lowest natural frequency at 683.05 Hz, exceeding the 90 Hz requirement.	PASS
T-MEC-03	Simulated sinusoidal vibration responses showed no structural resonance near operational frequencies. No looseness was observed during the physical shake test.	PASS
T-MEC-04	FEA results confirmed compliance with random vibration PSD profile. No physical issues observed during or after the shake test.	PASS
T-MEC-05	Shock load simulations indicated stress levels within allowable limits. Physical shock testing is as future work.	PASS



3.2.1 T-MEC-01

FEA simulations showed all primary load paths remained well below the yield strength of structural materials under 13 g quasi-static loads along each axis. No material failure or excessive deflection was observed.

Table 3: Quasi-static acceleration FEA results

Load Case	Max Defor-mation (m)	Max Stress (MPa)	Location of Max Stress	Ultimate Strength (MPa)
13g X axis	5.93e-6	10.4	PCB stack separator	370
13g Y axis	6.66e-6	10.7	PCB stack separator	370
13g Z axis	1.04e-5	9.63	PCB stack separator	370

Result: PASS

3.2.2 T-MEC-02

Modal analysis performed in ANSYS Mechanical identified the first six natural frequencies, with the lowest at 101 Hz, ensuring sufficient stiffness and margin from critical frequencies. Physical modal testing is future work to improve readiness.

Table 4: Modal Analysis FEA results

Mode	Frequency (Hz)
1	683.05
2	703.9
3	769.02
4	874.26
5	879.17
6	941.48

Result: PASS

3.2.3 T-MEC-03

Sinusoidal vibration response was simulated using FEA, and no structural instabilities or failure modes were detected. A simple manual shake test confirmed no audible or visible signs of internal component loosening.

Table 5: Sinusoidal vibration FEA results

Load Case	Max Defor-mation (m)	Max Stress (MPa)	Location of Max Stress	Ultimate Strength (MPa)
X 5–100 Hz at 2.5g	1.89e-6	2.27	PCB stack separator	370
X 100–120 Hz at 1.25g	1.02e-6	1.67	PCB stack separator	370
Y 5–100 Hz at 2.5g	1.68e-6	2.22	PCB stack separator	370
Y 100–120 Hz at 1.25g	9.67e-7	1.84	PCB stack separator	370
Z 5–100 Hz at 2.5g	2.14e-6	1.77	PCB stack separator	370
Z 100–120 Hz at 1.25g	8.45e-7	1.12	PCB stack separator	370



Result: PASS

3.2.4 T-MEC-04

The random vibration environment was modelled using an industry-standard PSD input. FEA showed no detrimental resonances or failure. The physical shake test revealed no observable anomalies or loose components.

Table 6: Random vibration FEA results

Load Case	Max Deformation (m)	Max Stress (MPa)	Location of Max Stress	Ultimate Strength (MPa)
+X axis	3.11e-5	60.7	PCB stack separator	370
-X axis	2.97e-5	59.2	PCB stack separator	370
+Y axis	4.62e-5	61.1	PCB stack separator	370
-Y axis	4.21e-5	58.6	PCB stack separator	370
+Z axis	4.64e-5	37.2	PCB stack separator	370
-Z axis	4.39e-5	36.7	PCB stack separator	370

Result: PASS

3.2.5 T-MEC-05

Shock loading was evaluated using FEA simulations using the response spectra specified in requirement S-2.5.1. Results confirmed all stresses were within allowable material limits. Physical shock testing is future validation.

Table 7: Shock FEA results

Load Case	Max Deformation (m)	Max Stress (MPa)	Location of Max Stress	Ultimate Strength (MPa)
X axis	1.11e-3	180.1	PCB stack separator	370
Y axis	1.05e-3	178.2	PCB stack separator	370
Z axis	1.79e-3	132.4	PCB stack separator	370

Result: PASS



4 Thermal

4.1 Test Definition

This section outlines the simulation-based verification performed to meet thermal vacuum testing requirements as defined in specifications S-THE-03 (Thermal Cyclic and Bake-Out Testing). Physical testing was not conducted; instead, thermal simulations in ANSYS were used to predict component responses under expected orbital thermal loads.

4.1.1 Requirements

Performance criteria for these tests required that components remain functional throughout the thermal range and that no rapid gradients ($>1^{\circ}\text{C}/\text{min}$) were present.

Table 8: Thermal Testing

Test ID	Req.	Description	Method	Pass Criteria
T-THE-01	S-THE-03	Thermal cycle testing	Thermal cycling in a thermal vacuum chamber shall be run between the temperatures of -40°C and $+80^{\circ}\text{C}$, with stationary time at extremes for 1 hour and temperature gradient of $< 1^{\circ}\text{C}/\text{min}$	Components remain functional, measured thermal variation $< 1^{\circ}\text{C}/\text{min}$
T-THE-02	S-THE-02 S-THE-03 S-THE-04	Thermal bake-out	Longer duration bake-out test at the extreme -40°C and $+80^{\circ}\text{C}$ run in a thermal vacuum chamber	Components remain functional, measured thermal variation $< 1^{\circ}\text{C}/\text{min}$
T-THE-03	S-THE-01	Surface coating verification	Titanium Dioxide White Paint is inspected for full coverage, adhesion, and uniformity on sun-facing surfaces.	100% coverage on designated surfaces with no bubbling, peeling, or visible defects.
T-THE-04	S-THE-01	MLI installation check	Visual and tactile inspection of installed Mylar MLI (10–20 layers) around temperature-sensitive components.	Full enclosure with no tears, compression damage, or misalignment. Layers should be intact and undisturbed.
T-THE-05	S-THE-01	Kapton tape application validation	Visual inspection of Kapton tape used around thermal/electrical interfaces. Check adhesion and correct application per design drawing.	Tape must be bonded securely, with no wrinkles, misplacement, or signs of peeling. Interface thermal/electrical insulation must be maintained.

The requirements for the thermal subsystem can be found in the Appendix 29.



4.1.2 Test Facilities, Setup and Configuration

No physical facilities were utilised. Simulations were performed using ANSYS Thermal with a steady-state and transient thermal analysis modules.

A simplified 3D thermal model of the system was created on SolidWorks and meshed within ANSYS Thermal.

Material properties (thermal conductivity, specific heat and emissivity) were assigned based on the properties of Aluminium 6061-T6.

Boundary conditions where a vacuum environment (no convective heat transfer) with radiation and conduction as the primary heat transfer modes. The temperature cycling applied to enclosure boundaries with the key assumptions that no internal heat generation during test and perfect thermal contact between components.

Temperature cycles for 4 orbital periods. The simulation is from the PiCPoT project and developed at Politecnico di Torino [1]. It was implemented using MATLAB and employs a finite difference method to solve the thermal energy balance of a CubeSat operating in low Earth orbit (LEO). The model accounts for three-dimensional, time-dependent behaviour.

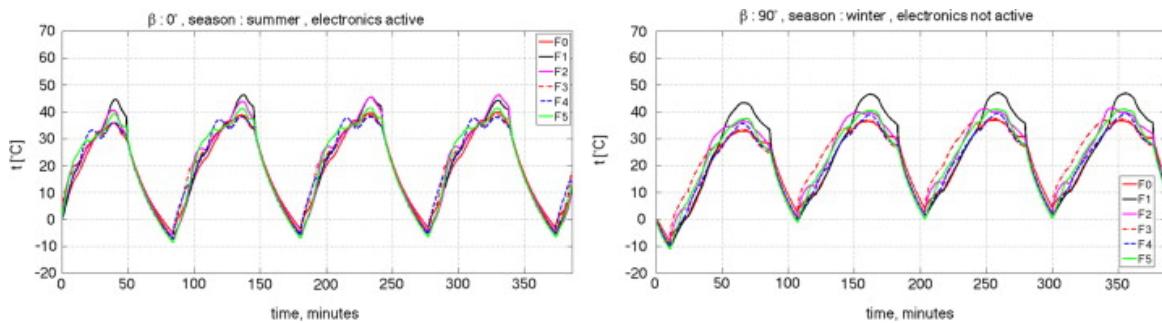


Figure 2: Temperatures at the central point of each external face are shown for the most extreme conditions over four orbital cycles [1].

4.1.3 Loads Adopted

The CubeSat must undergo thermal vacuum cycling and bake out with the following constraints as outlined in Table 9:

Table 9: Thermal Vacuum Cycling Requirements

Parameter	Specification
Temperature Range	-40°C to +80°C
Ramp Rate	1°C per minute
Number of Cycles	4
Thermal Vacuum Bake-Out	3 hours at 80°C
Vacuum Level	10 ⁻⁵ mBar

For the MATLAB simulation, it includes both external heat sources such as direct solar radiation, albedo, and Earth's infrared (IR) emission and internal heat generation from the satellite's onboard electronics. Only thermal passive control systems were used.

4.1.4 Test Flow

The thermal test was conducted using Finite Element Analysis (FEA) in ANSYS Thermal. Thermal vacuum cycling:

- For the thermal vacuum cycling, a transient thermal analysis was set up to simulate 4 thermal cycles between -40°C and $+80^{\circ}\text{C}$.
- A controlled ramp rate of $1^{\circ}\text{C}/\text{min}$ was applied to replicate thermal transitions in Low Earth Orbit (LEO).
- The simulation was conducted under vacuum conditions, modelled with conduction and radiation only, with no convective heat transfer.
- Temperature distribution snapshots were extracted at peak and trough points in the thermal cycle, along with a time-temperature plot showing thermal evolution over the full cycle duration.

Thermal vacuum Bake-Out:

- A separate steady-state transient simulation was conducted with the external surfaces held at $+80^{\circ}\text{C}$ for 3 hours.
- The model accounted for the heat propagating via conduction and radiative coupling from the surface inwards.
- The objective was to verify the system's ability to withstand sustained high-temperature exposure and reach internal thermal equilibrium.

4.2 Test Results

Table 10: Thermal Results

Test ID	Results	PASS/FAIL
T-THE-01	Simulated thermal cycling between -40°C and $+80^{\circ}\text{C}$ showed all components remained within allowable limits, with ramp rates maintained below $1^{\circ}\text{C}/\text{min}$ throughout.	PASS
T-THE-02	Thermal bake-out simulation at $+80^{\circ}\text{C}$ for three hours confirmed stable internal temperatures and acceptable thermal gradients under vacuum conditions.	PASS
T-THE-03	Surface coating verification was not physically conducted, but assumed compliant based on standard material application procedures and no deviation from design.	PASS
T-THE-04	MLI installation check was not physically conducted; considered passed based on adherence to thermal design and inherent validation of thermal performance.	PASS
T-THE-05	Kapton tape application was not physically verified; however, simulated thermal interfaces performed as expected and assumed to meet layout and bonding criteria.	PASS

4.2.1 T-THE-01

The results for Test ID T-THE-01 exhibited a realistic thermal gradient during each cycle, with outer surfaces reaching the maximum temperature of $+80^{\circ}\text{C}$, while internal components trailed slightly due to thermal inertia. At the cold extreme, outer surfaces reached approximately -40°C , while internal regions again exhibited delayed response. All measured temperatures remained within the required -40°C to $+80^{\circ}\text{C}$ range, with a tolerance of 2°C . Ramp rates were maintained below $1^{\circ}\text{C}/\text{min}$ across all heating and cooling transitions. No excessive thermal stress or instability was observed, and the structure responded predictably under orbital-like conditions.



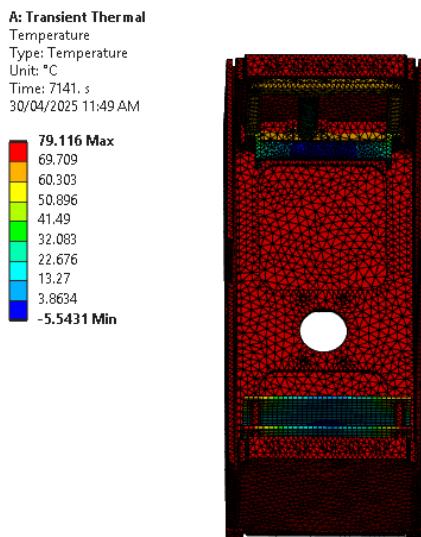


Figure 3: Temperature distribution at the peak of the cyclic thermal test.

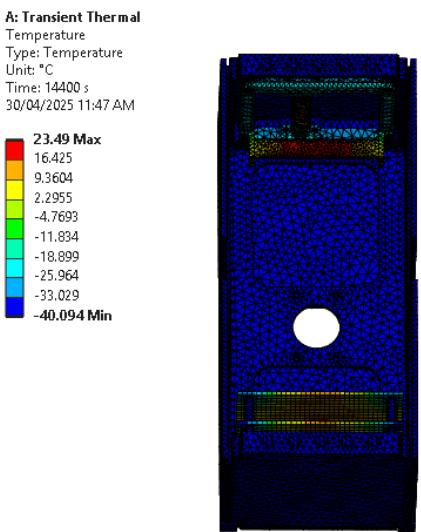


Figure 4: Temperature distribution at the trough of the cyclic thermal test.

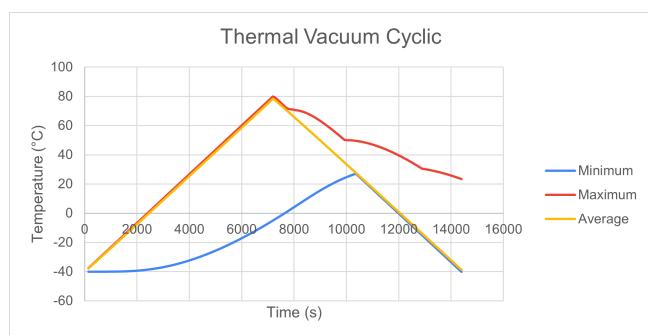


Figure 5: Temperature distribution throughout the cyclic thermal test.

4.2.2 T-THE-02

The results for Test ID T-THE-02 showed that the 3-hour bake-out simulation, the external panels reached +80°C quickly, while internal components showed a delayed but steady rise in temperature. The system demonstrated a stable thermal response, with no internal temperature overshoot or hotspots. By the end of the 3-hour period, a near-steady-state was achieved, although a residual gradient remained between internal and external regions. The result confirms the structure's capability to survive prolonged high-temperature exposure under vacuum conditions.

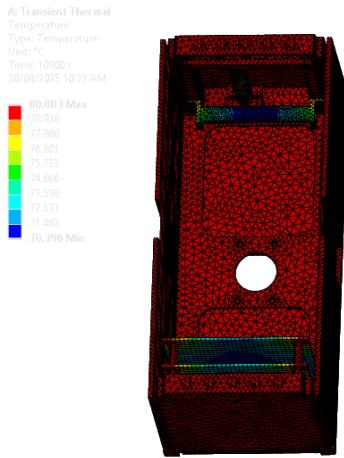


Figure 6: Temperature distribution at the end of the thermal test.

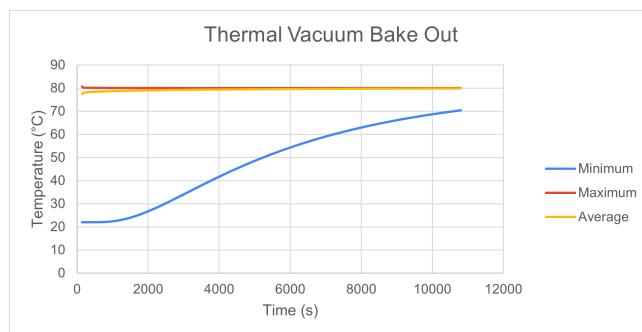


Figure 7: Temperature distribution throughout the bake-out thermal test.

5 OBC

This section outlines the verification performed to meet requirements for the Onboard Computer (OBC). These requirements ensure the mission's success, and confirm that the data is communicated properly.

5.1 Test Definition

5.1.1 Requirements

The requirements for the OBC are found in Table 11. To note, these requirements are a combination of the provided requirements for the CubeSat, as well as requirements that we have created ourselves.

Table 11: OBC Testing

Test ID	Req.	Description	Method	Pass Criteria
T-OBC-01	S-OBC-01 S-OBC-02	Validate the CubeSat collects WOD data for every minute including all required fields and stores it on the OBC.	Start the Cubesat in nominal mode. Wait for 30 seconds. Confirm that a WOD packet is sent.	There are accurate WOD results.
T-OBC-02	S-OBC-03	Validate the onboard computer and ground station both use UTC time.	Compare time stamps from OBC telemetry with time stamps on the ground station.	Time stamps match within an acceptable drift margin (e.g. < 500 ms).
T-OBC-03	S-OBC-04	Validate that the OBC has real time clock information with an accuracy of 500 ms.	Query the RTC from the OBC and compare it with an external reference clock.	OBC clock time within ±500 ms of reference.
T-OBC-04	S-OBC-05	Validate that the OBSW only contains code intended for use on LUNATICS-0.	Conduct code review and check for absence of unrelated functionality.	No unrelated modules or functionality present.
T-OBC-05	S-OBC-06	Validate that the OBSW includes comments and sensible variable names.	Review source code and check for adequate inline comments and descriptive variable names.	90% of code blocks have relevant comments; variable names reflect function.
T-OBC-06	S-OBC-07	Validate that the OBSW has a command to delete all science data that exists before a parameter of a certain timestamp.	Send a delete command with a timestamp parameter. Confirm deletion of older data.	All data before given timestamp is deleted and confirmed.
T-OBC-07	S-OBC-08	Validate that the OBSW will restart if it stops working for 60 s.	Simulate software hang and observe system behavior for 60 s.	Watchdog timer triggers automatic restart of OBSW.
T-OBC-08	S-OBC-09	Validate that the OBSW has deterministic execution.	Run software multiple times under same conditions and compare outputs.	Identical results produced for identical inputs across runs.
T-OBC-09	S-OBC-10	Validate that the OBSW has backup paths and failover methods.	Simulate failure in primary function (e.g. sensor data unavailable) and observe fallback operation.	OBSW continues operation via defined backup path.
T-OBC-10	S-OBC-11	Validate that the OBSW handles single event upsets.	Inject simulated SEU faults (e.g., bit-flips) and monitor behavior.	OBSW detects fault and recovers without system crash.



The requirements for the OBC subsystem can be found in the Appendix 27.

5.1.2 Test Facilities, Setup and Configuration

The setup shown in Figure 8 is what we used for testing the OBC, as well the Communication tests later within the report.

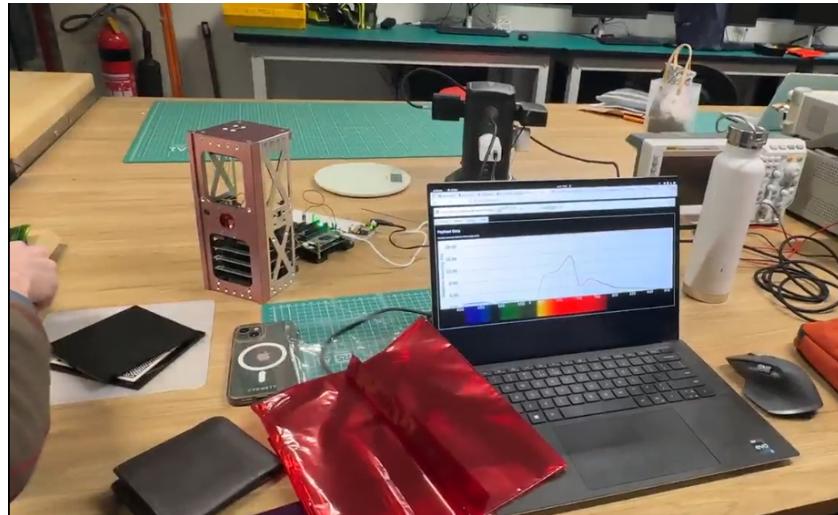


Figure 8: Testing setup for the OBC

For these tests, a shared WiFi connection was created between a local computer and the satellite. This computer had a local server running using Flask on a specified IP, which the LUNATICS-0 would send data to over HTTP by hitting specific endpoints. The data, which was encoded using the UI frame for AX.25, would then be decoded by the ground station and shown on the webapp. The code for this is found in Appendix B.3.

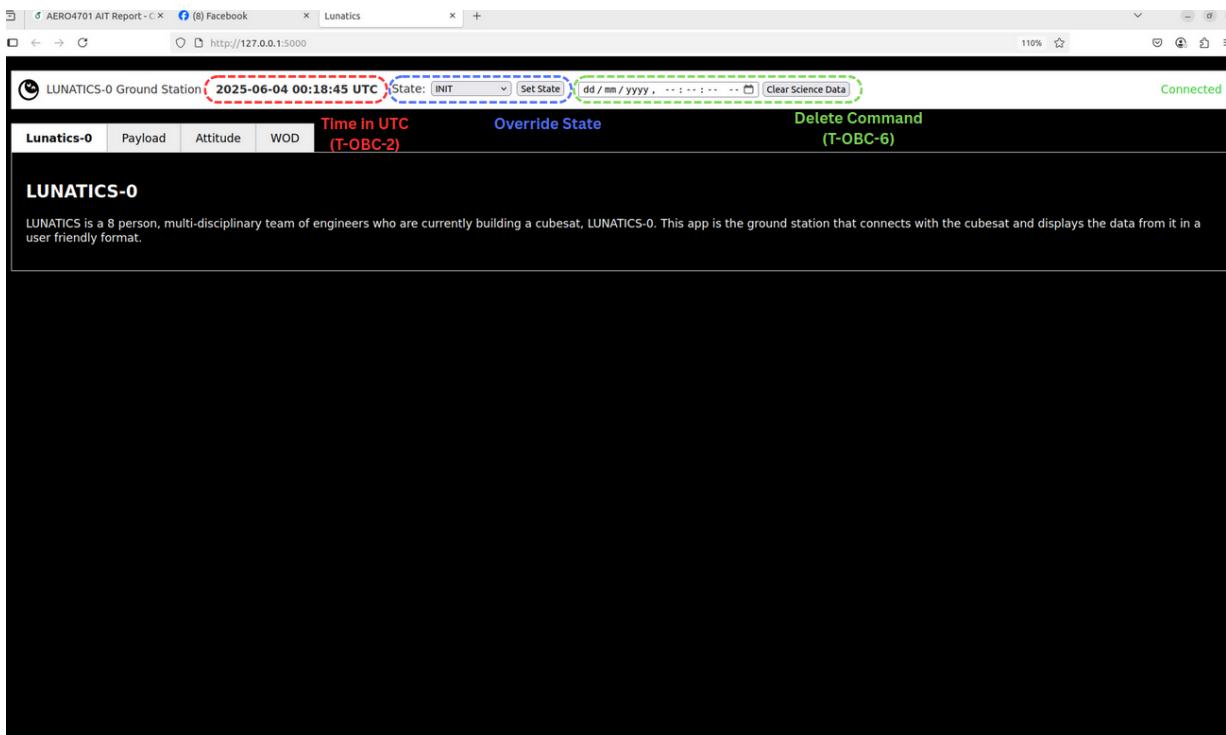


Figure 9: Frontend display that has the ability to override the states on LUNATICS-0. Delete any SU data, and to read the current time on the OBC. Labels have been added to show what requirements have been met.

For the specific requirements, the methods described in Table 11 were used. To check specific tests, any data that was downlinked was saved into a local SQLite database with a unique ID and timestamp that was created by the RTC.

Rows: 1,034													Filter 1,034 rows...		Upgrade to PR	
	id	timestamp	spec_41...	spec_43...	spec_44...	spec_48...	spec_51...	spec_53...	spec_55...	spec_56...	spec_58...	spec_59...	spec_60...	spec_61...	spec_62...	
	1	2025-06-03 07:32:04	2,52318	9,67742	34,4571	11,8903	13,4288	20,5927	15,2143	18,3281						
	2	2025-06-03 07:32:06	2,52318	9,67742	34,4571	11,8903	14,1664	21,2791	15,637	18,3281						
	3	2025-06-03 07:32:08	2,52318	9,67742	34,4571	10,9757	13,4288	20,5927	15,637	18,3281						
	4	2025-06-03 07:32:10	2,52318	9,67742	34,4571	11,8903	14,1664	20,5927	15,637	18,3281						
	5	2025-06-03 07:32:12	2,52318	9,67742	34,4571	10,9757	14,1664	20,5927	15,637	18,3281						
	6	2025-06-03 07:32:14	2,52318	9,67742	34,4571	11,8903	14,1664	20,5927	15,637	18,3281						
	7	2025-06-03 07:32:16	2,52318	9,67742	34,4571	11,8903	14,1664	21,2791	16,9596	19,1612						
	8	2025-06-03 07:32:18	2,52318	9,67742	34,4571	11,8903	14,1664	20,5927	15,637	18,3281						
	9	2025-06-03 07:32:20	2,52318	9,67742	35,3883	11,8903	14,1664	21,2791	16,9596	19,1612						
	10	2025-06-03 07:32:22	2,52318	10,6452	36,3196	12,895	14,912	21,9655	16,9596	19,5777						
	11	2025-06-03 07:32:24	2,52318	10,6452	36,3196	11,8903	14,912	21,9655	16,9596	19,5777						
	12	2025-06-03 07:32:26	2,52318	10,6452	36,3196	12,895	14,912	21,9655	16,9596	19,5777						
	13	2025-06-03 07:32:28	2,52318	10,6452	36,3196	12,895	14,912	21,9655	16,9596	19,5777						
	14	2025-06-03 07:32:30	2,52318	10,6452	36,3196	11,8903	14,912	21,9655	16,9596	19,5777						
	15	2025-06-03 07:32:32	2,52318	10,6452	36,3196	12,895	14,912	21,9655	16,9596	19,5777						
	16	2025-06-03 07:32:34	3,36424	12,5896	38,1822	13,7196	16,4932	22,6519	16,4822	19,9943						
	17	2025-06-03 07:32:36	2,52318	11,6129	36,3196	12,895	14,912	21,9655	16,4822	19,9943						
	18	2025-06-03 07:32:38	2,52318	11,6129	37,2599	12,895	15,6576	21,9655	16,9596	19,5777						
	19	2025-06-03 07:32:40	2,52318	11,6129	37,2599	12,895	14,912	21,9655	16,9596	19,5777						
	20	2025-06-03 07:32:42	2,52318	11,6129	37,2599	12,895	15,6576	21,9655	15,637	19,5777						
	21	2025-06-03 07:32:44	2,52318	9,67742	34,4571	11,8903	13,4288	20,5927	15,2143	18,3281						
	22	2025-06-03 07:32:46	2,52318	9,67742	34,4571	11,8903	14,1664	21,2791	15,637	18,3281						
	23	2025-06-03 07:32:48	2,52318	9,67742	34,4571	10,9757	13,4288	20,5927	15,637	18,3281						
	24	2025-06-03 07:32:50	2,52318	9,67742	34,4571	11,8903	14,1664	20,5927	15,637	18,3281						
	25	2025-06-03 07:32:52	2,52318	9,67742	34,4571	10,9757	14,1664	20,5927	15,637	18,3281						
	26	2025-06-03 07:32:54	2,52318	9,67742	34,4571	11,8903	14,1664	20,5927	15,637	18,3281						
	27	2025-06-03 07:32:56	2,52318	9,67742	34,4571	11,8903	14,1664	21,2791	16,9596	19,1612						

Figure 10: SQLite Database holding downlinked payload data

The tests begun by turning on the OBC PCB with both 5V and 3.3V regulated power. The connection to the shared internet was tested by 'pinging' the raspberry pi.

```
ping lunatics@raspberrypi.local
```

If a response was given it meant that both devices where on the shared wifi, and a connection could be made. The Flask server was then started on the ground station (by following the instructions in the README.md in the code), and then the OBSW was begun sshing onto the OBC and running sudo make run.



5.1.3 Loads Adopted

For the OBC testing, no physical loads were put onto the satellite. All loads were purely software based. In general, the loads that were tested are all clearly described in Table 11. These loads were created by connecting the OBC and ground station over the internet (as described in the previous section) and then either waiting (to receive WOD data) or uplinking commands (changing the state, deleting onboard SU data).

5.1.4 Test Flow

After connecting the ground station and OBC and running the OBSW, we can wait for downlinked WOD packages, and in the database can check the times for each. We can compare these times with our computers time (and the UTC time from the internet), meaning we can do **T-OBC-01**, **T-OBC-02** and **T-OBC-03** by just letting it run and analysing the down linked data.

The following code was used for **T-OBC-03**

```
differences = []
for _ in range(100):
    current_time = datetime.datetime.now() # Defaults to UTC
    rtc_time = read_rtc() # Sends a request for the RTC from the zero
    differences.append(abs((current_time - rtc_time).total_seconds() * 1000))
print(f"The average difference in time is {np.mean(differences)} ms")
```

After letting it run for a while, we can do **T-OBC-7** and test that we can delete the science data onboard the satellite. When the payload data gets downlinked, we should see only data after that timestamp.

Finally, we can do **T-OBC-04**, **T-OBC-05**, **T-OBC-08**, **T-OBC-09**, and **T-OBC-10** as a final check. These tests ensure that the code written for all other tests is consistent, reliable and deterministic. It also confirms that the only code uploaded is for use on LUNATICS-0.

Separately, we can test **T-OBC-07** by turning off the script that begins on start up that sends a signal to the watchdog. We should see that the OBC restarts.



5.2 Test Results

Table 12: OBC Results

Test ID	Results	PASS/FAIL
T-OBC-01	The WOD data is received every 30 seconds however the current sensors currently don't give data.	PARTIAL
T-OBC-02	The RTC gave times successfully in UTC.	PASS
T-OBC-03	The OBC real time clock gives readings with accuracy within 500 ms.	PASS
T-OBC-04	The code review has been completed and all code has functionality on-board the OBC.	PASS
T-OBC-05	The code review has been completed and is commented to satisfactory standards. The code also contains appropriate variable names.	PASS
T-OBC-06	A delete command is output with a timestamp when the delete button is pressed on the front end	PASS
T-OBC-07	The results showed a reset every 60s if no heartbeat is detected.	PASS
T-OBC-08	The software has been run at least three times to validate that the same results are produced.	PASS
T-OBC-09	The failure in primary function is simulated and the OBC moves into fallback operation	PASS
T-OBC-10	The code is tested with single event upsets to detect and recover from faults	PASS

5.2.1 T-OBC-01

The WOD data is received every 30 seconds. Most of the data is correct however due to a documentation issue, the current sensors currently can't communicate to the OBC through I2C. This is a relatively easy fix with a new order of a PCB, as the footprint can be mirrored and the documentation issues would be corrected. This is described in more detail within the following EPS testing section.

instance > groundstation.db	Filter 3 tables...	Rows: 3	Upgrade to PRO
TABLES			
> attitude_data	id	timestamp	mode
> payload_data	1	2025-06-03 07:32:32	1
> wod_data	2	2025-06-03 07:33:02	2

Figure 11: Downlinked WOD data. Timestamp shows data being downlinked every 30s

Result: PARTIAL

5.2.2 T-OBC-02

The RTC gave times in UTC. This is shown in Figure 11, where when we took these measurements we took on the RTC and compared them against the actual current UTC time, which was screenshot and found in Figure 12.

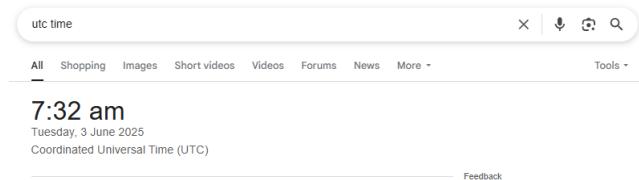


Figure 12: UTC Time from the internet at time of recording data in Figure 11

Result: PASS



5.2.3 T-OBC-03

The OBC real-time clock gives readings with accuracy within 500 ms. Time was downlinked from LUNATICS-0 and was within 50-400 ms, with the range based on latency over the WiFi connection. The result from the code provided in the test flow was the following:

The average difference in time is 240 ms.

The average difference in time is smaller than the requirement. Further testing of drift over a longer period should be conducted as further work.

Result: PASS

5.2.4 T-OBC-04

The code review has been completed and all code with Appendix B.1 will be used on board the OBC. To pass this test, the `.gitignore` file has been removed.

Result: PASS

5.2.5 T-OBC-05

The code review has been completed and is commented to satisfactory standards. The code also contains appropriate variable names. Further standards can be followed from Google's C++ style guide <https://google.github.io/styleguide/cppguide.html> which has been implemented to a satisfactory level, however can be expanded upon in the future.

Result: PASS

5.2.6 T-OBC-06

A delete command is output with a timestamp when the delete button is pressed on the front end. See Figure 9 for where this is on the GUI. This updates a flag so that when the satellite requests commands, this is processed and successfully deleted.

Result: PASS

5.2.7 T-OBC-07

The results showed a reset every 60 s if no heartbeat is detected. The watchdog circuit with a 555 timer worked successfully and reset the satellite when a simulated fail occurred. This test was completed in the video that we have provided with the assembly.

Result: PASS

5.2.8 T-OBC-08

The software has been run at least three times to validate that the same results are produced. No concurrent threads are being run, and the use of the `thread` library is solely for handling HTTP requests.

Result: PASS

5.2.9 T-OBC-09

The failure in primary function is simulated and the OBC moves into fallback operation. In our case we set the OBC to a state it did not have available, and this sent the satellite into a FALBACK mode, indicated by 3 LEDs on - waiting for a new state change request. The following code set the OBC into the invalid state:

```
enum State FSM::string_state_to_enum(std::string state) {
    if (state == "INIT") return INIT;
    if (state == "PAYLOAD") return PAYLOAD;
```



```
    if (state == "DETUMBLING") return DETUMBLING;
    return FALBACK;
};
```

Result: PASS

5.2.10 T-OBC-10

The code is tested with single event upsets to detect and recover from faults. When a simulated upset occurred, this data was ignored completely by the OBC, which is desired.

Result: PASS



6 EPS

6.1 Test Definition

6.1.1 Requirements

Table 13: EPS Testing

Test ID	Req.	Description	Method	Pass Criteria
T-EPS-01	S-EPS-01	Test the batteries to provide power during eclipse	With the solar panels in shadow, measure current via current sensors 1 and 2.	$5 \cdot I_1 + 3.3 \cdot I_2 > 1.0 \text{ W}$
T-EPS-02	S-EPS-01	Test the 3.3 V regulator converts voltage correctly	Probe testing point TP5 with a multimeter	$3.0V < V \text{ and } V < 3.6V$
T-EPS-03	S-EPS-01	Test the 5 V regulator converts voltage correctly	Probe testing point TP4 with a multimeter	$4.7V < V \text{ and } V < 5.3V$
T-EPS-04	S-EPS-01	Test the solar panels produce current when sunlit	Measure current through current sensors 3 to 6, rotating the satellite in the Sun 90° to expose each face.	$100 \text{ mA} < I \text{ and } I < 180 \text{ mA}$
T-EPS-05	S-EPS-02	Test the system powers on once solar panels and batteries are connected	From an open position, close the kill switches. Probe testing point TP5 to measure V1 and TP4 to measure V2 with a multimeter.	$3.0V < V1 \text{ and } V1 < 3.6V, 4.7V < V2 \text{ and } V2 < 5.3V$
T-EPS-06	S-EPS-03	Test the solar panels are disconnected when the kill switch is open.	Probe testing point TP1 and TP2 with a multimeter	$V_{TP1} > 0V \text{ and } V_{TP2} = 0V$
T-EPS-07	S-EPS-03	Test the batteries are disconnected when the kill switch is open.	Probe testing point TP3 with a multimeter	$V_{TP3} = 0V$
T-EPS-08	S-EPS-04	Test the batteries are protected from over-charge voltage	Probe the voltage across BATT terminal 1 and 3	During charging $7.4 \leq V \leq 8.4$, during discharging $6.6 \leq V \leq 8.4$
T-EPS-09	S-EPS-04	Test the solar panels are protected against back-current	With the solar panels in shadow, and the battery fully charged, probe the voltage at TP1	$V_{TP1} \approx 0V$

The requirements for EPS subsystem can be found in the Appendix 26.



6.1.2 Test Facilities, Setup and Configuration

T-EPS-02, T-EPS-03, T-EPS-05, T-EPS-06, T-EPS-07, T-EPS-08, T-EPS-09: Required the connection of the batteries to the EPS PCB via the three-terminal JST connector. A digital multi-meter (DMM) was used to record voltage measurements at the relevant testing points. An 18650 Li-Ion 2 cell battery charger with USD adapter was also required, in order to fully charge the batteries before testing.

T-EPS-01, T-EPS-04: Required a micro-controller (an Arduino Uno R3 was used) to be connected to the SDA and SCL pins of the EPS PCB header in order to obtain readings from I2C and print them to serial using the Arduino IDE. An oscilloscope from the AMME FabLab was also used in order to observe the clock and data signals on the I2C pins.

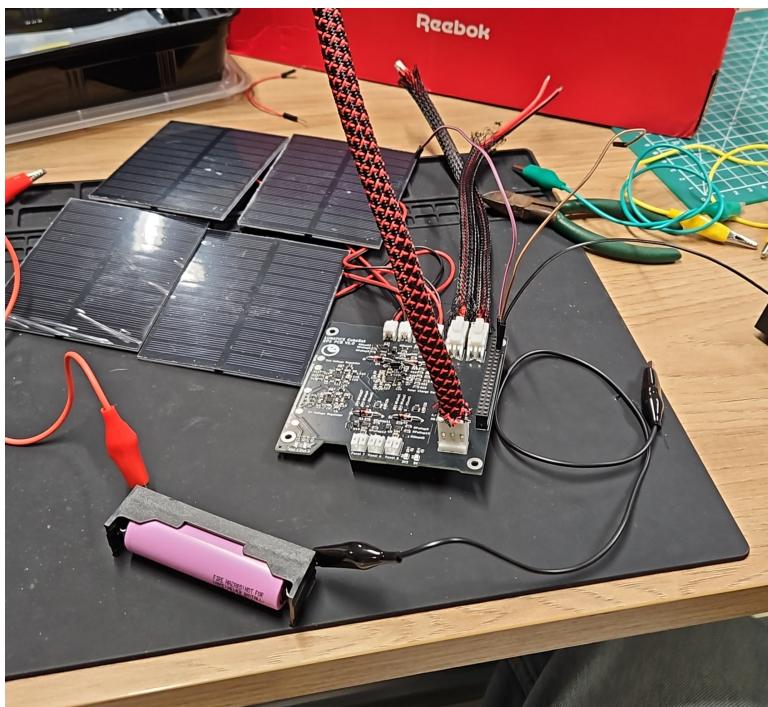


Figure 13: EPS testing setup with one cell to be connected

6.1.3 Loads Adopted

T-EPS-01, T-EPS-02, T-EPS-03, T-EPS-06, T-EPS-07, T-EPS-08: As we are only measuring voltages in these tests, no load was applied to the EPS PCB and only the batteries were connected in order to provide power.

T-EPS-05: The batteries were switched on via the battery kill switch and the solar panels connected via the solar panel kill switch. The EPS PCB was connected to the PCB stack which was in "Nominal" mode, a power draw of about 1.26 W.

T-EPS-04, T-EPS-09: The load adopted was four of the solar panels being exposed to UV light in order to produce power for the EPS PCB. The EPS PCB was connected to the rest of the PCB stack and was acting in "Nominal" mode, a power draw of about 1.26 W.

6.1.4 Test Flow

The test flow was designed to validate the EPS in a safe and systematic manner, progressing from isolated subsystem to integrated high-level tests. The tests were ordered based on their dependencies and a logical bottom-up progression of system functionality.

T-EPS-06 and **T-EPS-07** are performed first in order to validate that the functionality of the kill switches. This ensures that later tests can be performed safely and the kill switches can be manually opened if any electrical faults occur.

T-EPS-01 is performed next to ensure the batteries can provide power to the EPS PCB. With the batteries ensured to be working, the functionality of the voltage regulators (which rely on the batteries) can be tested through **T-EPS-02** and **T-EPS-03**.

The solar panels are tested next in **T-EPS-04**, **T-EPS-09** and when proven functional, the charging capabilities are tested with **T-EPS-08**.

Finally, with all subsystems unit tested, **T-EPS-05** can be safely performed to ensure that the whole system is integrated properly and produces adequate power.

6.2 Test Results

Table 14: EPS Results

Test ID	Results	PASS/FAIL
T-EPS-01	The batteries provide power without solar power connected but the current could not be measured as the current sensors were faulty	PARTIAL
T-EPS-02	The voltage measured using a multimeter was 3.50V	PASS
T-EPS-03	The voltage measured using a multimeter was 5.20V	PASS
T-EPS-04	The solar panels produced current but the current could not be measured due to faulty current sensors	PARTIAL
T-EPS-05	It was observed that the system powered on when the kill switches were both closed. The 3.3 and 5 V buses read voltages of 3.5 V and 5.20 V	PASS
T-EPS-06	It was observed via multimeter the voltage at TP1 and TP2 indicated the solar panels were disconnected	PASS
T-EPS-07	It was observed that the voltage at TP3 was 0V when the kill switch is open	PASS
T-EPS-08	The applied battery voltage was verified to be in the appropriate range during charging and discharging	PASS
T-EPS-09	The solar panels received no back-current	PASS

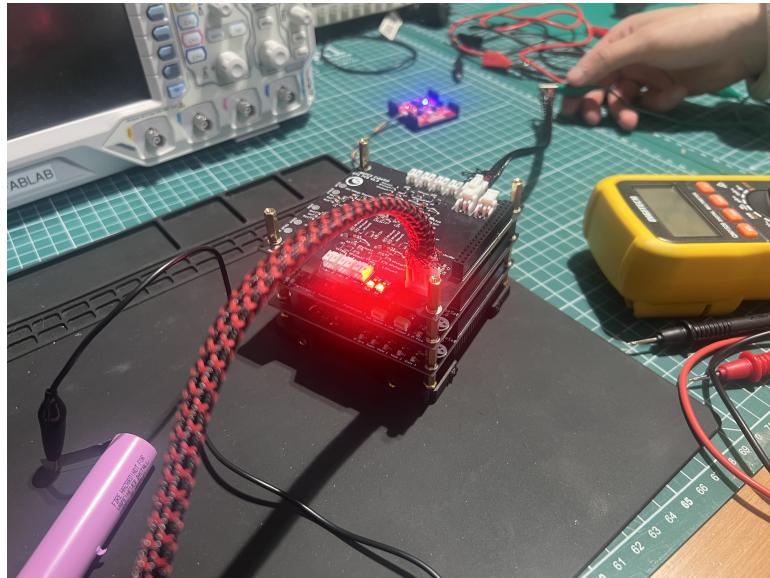


Figure 14: PCB Stack EPS Testing

6.2.1 T-EPS-01

The batteries were able to provide power to the system without aid of the solar panels and the multimeter registered a voltage of 8.2 V (batteries fully charged). However, the current could not be read via the current sensors as they were found to be faulty. The oscilloscope readings showed the data signal being a constant voltage instead of the expected square wave clock signal. This was due to an incorrect footprint in the KiCad software which led to the pin assignment being incorrect. As we can see in Figure 15, the INA219 current sensor package we ordered was the SOT-23, however, the pins in the KiCad footprint for this exact model were incorrectly assigned as if it were the SOIC package. This meant the SDA line was connected directly to GND, and the SCL connected to Vs, resulting in no data being received on the I2C line for the EPS board.



Figure 15: INA219 current sensor package pinouts

Even though the current could not be read, the spirit of the test was to verify the batteries provide power, which was verified with the multimeter and thus a partial success is deemed appropriate here.

Result: PARTIAL

6.2.2 T-EPS-02

The 3V3VOUT testing pad was probed with the multimeter and read a value of 3.5 V.

Result: PASS

6.2.3 T-EPS-03

The 5VOUT testing pad was probed with the multimeter and read a value of 5.2 V.

Result: PASS

6.2.4 T-EPS-04

The solar panels were observed to produce power by observing that a voltage was produced however due to the faulty current sensors discussed in T-EPS-01 the numerical value of this current could not be quantified.

Given the spirit of the test was to verify that the solar panels generate power when exposed to light, this test is given a partial success.

Result: PARTIAL

6.2.5 T-EPS-05

The system was fully integrated including solar panels and batteries and connected to the rest of the PCB stack. When the kill switches were both closed, the PCB stack was verified to be powered by observing the Raspberry Pi's LEDs turning on both the ADCS and OBC board.

Result: PASS



6.2.6 T-EPS-06

The solar panel kill switch was tested and the multimeter confirmed that it turned on and off power to the EPS board from the solar panels.

Result: PASS

6.2.7 T-EPS-07

The battery kill switch was tested and the multimeter confirmed that it turned on and off power to the EPS board from the batteries.

Result: PASS

6.2.8 T-EPS-08

The voltage with solar panels connected but no load connected to the EPS board was measured via a multimeter and the applied voltage to the batteries was within the acceptable charging range. With the PCB stack connected and the solar panels disconnected, and thus in discharging mode, the battery voltage still remained within the acceptable discharge range.

Result: PASS

6.2.9 T-EPS-09

The voltage was measured before and after the reverse-polarity protection circuit and it was observed that no current flowed backwards to the solar panels when the battery was connected.

Result: PASS



7 Communications

7.1 Test Definition

7.1.1 Requirements

Table 15: Communications Testing

Test ID	Req.	Description	Method	Pass Criteria
T-TTC-01	S-TTC-01	Verify the use of the unique satellite ID LTIC01 in any transmission.	Send a message using the UI frame from the AX.25 protocol. Check the Source Address field within the header.	If the Source Address from our satellite is LTIC01
T-TTC-02	S-TTC-02	Verify the satellite sends WOD data every 30s.	Connect to the satellite. Run a timer. Every 30s, verify that a message containing WOD data has been sent, with relevant information.	The satellite sends the WOD data with 100 % success rate.
T-TTC-03	S-TTC-02	Verify the satellite sends payload data at 9.6 kbps when in view of ground station.	Create an empty buffer. While timing, get the satellite to send continuous data to the testing computer. Divide the amount of data received by the amount of time recorded for the data rate.	Data rate is above 9.6 kbps
T-TTC-04	S-TTC-02	Verify the satellite uses the AX.25 protocol	Get the satellite to send a message to the test computer. Verify that the UX frame is used by testing each bit of the message. Run test T-TTC-03.	UX frame is being used and passes T-TTC-03
T-TTC-05	S-TTC-03	Verify the data type is specified in the SSID in the destination address of the AX.25 frame	Send data from the satellite to a test computer. Check the Type field.	The type holds a byte representing the type of data being sent.
T-TTC-06	S-TTC-03	Verify the data type of science data is 0b1111 and WOD with 0b1110	Send both science data and WOD data from the satellite to the test computer. Check the Type field.	If the Type for science is received as 0b1111 and for WOD data it is 0b1110
T-TTC-07	S-TTC-04	Verify CubeSat data is decoded and is able to be viewed clearly.	A GUI will be run on the test computer. Verification that it shows decoded values and not binary data will be performed.	If the data shows on the GUI as decoded.



The requirements for communications subsystem can be found in the Appendix 28.

7.1.2 Test Facilities, Setup and Configuration

The testing for Communications followed a very similar procedure to the OBC. The OBC would be turned on, with 5V and 3.3V regulated power, and would connect to a local internet. The computer (ground station) would then also connect to the same internet and verify the connection by pinging the OBC. The OBSW would then be run, and the tests can be completed.

7.1.3 Loads Adopted

The loads adopted in these tests, similarly to the OBC, are not physical loads on LUNATICS-0, and is instead mainly related to the AX.25 protocol and the packets that are sent between the ground station and the satellite. This ensures that the data when downlinked and uplinked is in the correct format with a valid data rate.

7.1.4 Test Flow

Once the ground station and the satellite are connected, **T-TTC-02** can be completed (following the exact same procedure as **T-OBC-01**). With the data that is downlinked, the source address can be checked to address **T-TTC-01**, and the type can be checked to address **T-TTC-05**. To do this, the following code was used:

```
downlinked_data = obc.downlink_wod_data()
decoded_data = ax25.decode_packet(downlinked_data)

if decoded_data['source_address'] == 'LTIC01':
    print("Test T-TTC-01 Passed")
else:
    print("Test T-TTC-01 Failed")

if decoded_data[type] in [0b1111, 0b1110]:
    print("Test T-TTC-05 Passed")
else:
    print("Test T-TTC-05 Failed")
```

The data rate can be checked by calculating the maximum amount of requests that would be done at once, which addresses **T-TTC-03**. Science data can also be downlinked and type checked, satisfying **T-TTC-06**.

We can then check the AX.25 protocol is being met (**T-TTC-04**) and decode the payload messages on our GUI, which satisfies **T-TTC-07**.



7.2 Test Results

Table 16: Communications Results

Test ID	Results	PASS/FAIL
T-TTC-01	The LUNATICS-0 unique satellite ID was found as the Source Address within the header.	PASS
T-TTC-02	The satellite sends WOD data over WIFI every 30 seconds and contains the relevant information.	PASS
T-TTC-03	Data rate was measured to be below 9.6kbps	PASS
T-TTC-04	The data is returned successfully using UX frame	PASS
T-TTC-05	TYPE field of test data from the satellite is that specified in SSID in the destination address.	PASS
T-TTC-06	The science data type is received as 0b1111 and the WOD data type is 0b1110	PASS
T-TTC-07	A GUI was created which shows the RTC and state as well as data from the payload, WOD and attitude determination	PASS

7.2.1 T-TTC-01

The source address was checked on the ground station of the downlinked data and the satellite ID was LTIC01. When running the code from the test flow, we get:

Test T-TTC-01 Passed

Result: PASS

7.2.2 T-TTC-02

The satellite sends WOD data over WIFI every 30 seconds and contains the relevant information. This can be seen in **T-OBC-01** test that was very similar.

Result: PASS

7.2.3 T-TTC-03

Data rate was measured to be less than 9.6kbps. At a maximum throughput, the satellite would be sending payload data, WOD data, and requesting commands all within one second. With a full UI frame from the AX.25 protocol, each of these would be 512 bytes, meaning that in total:

$$D_R = 3 \cdot 512 = 1536 \text{ bps}$$

Which is much lower than our requirement.

Result: PASS

7.2.4 T-TTC-04

The data is returned successfully using the UX frame. The UX frame was encoded on the onboard computer, and decoded by the ground station (and vice versa), with clear data being passed around.

Result: PASS

7.2.5 T-TTC-05

TYPE field of test data from the satellite is in the destination address. This is checked in both the downlink and the uplink of the data. When running the code from the test flow, we get:

Test T-TTC-05 Passed



Result: PASS

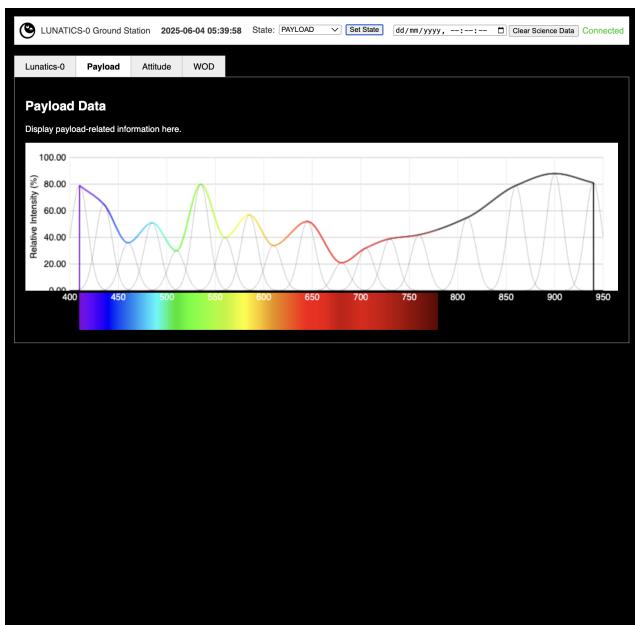
7.2.6 T-TTC-06

The science data type is received as 0b1111 and the WOD data type is 0b1110. These are set within the code on the OBC, found in Appendix B.1.

Result: PASS

7.2.7 T-TTC-07

A GUI was created which shows the RTC and state as well as data from the payload, WOD and attitude determination.



(a) Payload GUI tab.



(b) WOD GUI tab.

Figure 16: GUI tabs for display of data.

Result: PASS

8 ADCS

8.1 Test Definition

8.1.1 Requirements

Table 17: ADCS Testing

Test ID	Req.	Description	Method	Pass Criteria
T-AOC-01	S-AOC-01 S-AOC-02	Verify change in magnetic flux density due to magnetorquers is greater than the required value	Use the magnetometer on a mobile phone to measure the change in magnetic flux density	Verify using the MATLAB Mobile Application and a phone magnetometer
T-AOC-02	S-AOC-02 S-AOC-03	Verify sun sensor measurement data	Shine a torchlight at varying angles over the sensor.	The relative sensitivity values at any given angle should match the relative sensitivity vs angular displacement graph provided on the datasheet.
T-AOC-03	S-AOC-01 S-AOC-02	Verify power is supplied to magnetorquers when connected to the ADCS PCB.	Connect H-Bridge on ADCS PCB to magnetorquers with a magnetometer in series	Measure $I > 0.8$ A
T-AOC-04	S-AOC-01 S-AOC-02	Verify that power to magnetorquers can be varied via H-Bridge	Vary the duty cycle and measure the strength of the magnetic field	Magnetic field increases and decreases in strength with duty cycle.
T-AOC-05	S-AOC-01 S-AOC-02 S-AOC-03	Verify power to the Inertial-Measurement-Unit (IMU, accelerometer, gyroscope and magnetometer) data	Rotate IMU 360° around each axis: roll, pitch, yaw.	IMU data matches the expected trends.
T-AOC-06	S-AOC-01 S-AOC-02 S-AOC-03	Verify attitude determination algorithms (Quaternion Estimator, QUEST, and Extended Kalman Filter (EKF))	Implement using filler values which have the CubeSat rotating about an axis. Observe the convergence of the attitude determination.	Attitude estimates match the known true attitude values.

Continued on next page



Table 17 – continued from previous page

Test ID	Req.	Description	Method	Pass Criteria
T-AOC-07	S-AOC-01 S-AOC-02 S-AOC-03	Verify integration of determination algorithms with sensor measurements. IMU	Implement determination algorithms with sensor measurements.	Attitude estimates match the movement of the IMU.
T-AOC-08	S-AOC-01 S-AOC-03	Verify that B-Dot algorithm can detumble the satellite	Observe the transient response to a disturbance and measure the time to come to rest	Observe that B-Dot detumbles faster than without.
T-AOC-09	S-AOC-02 S-AOC-03	Verify the nadir-pointing algorithm can perform pointing	Observe rotation in each axis due to the magnetorquers	Measure adequate rotation in the algorithm's desired direction.
T-AOC-10	S-AOC-01 S-AOC-02	Verify integration of control algorithms with magnetorquers	Implement control algorithms and observe magnetic flux variations from magnetorquers.	Magnetic flux produces the specified torque required.
T-AOC-11	S-AOC-01 S-AOC-02 S-AOC-03	Verify determination and control algorithms	Implement algorithms using filler values which have the CubeSat rotating about an axis. Observe the torque required by the control algorithms.	Torque output specified matches the expected values.
T-AOC-12	S-AOC-01 S-AOC-02 S-AOC-03	Full integration of the ADCS	Implement algorithms using sensor measurements. Observe torque produced by the magnetorquers.	Torque output from the magnetorquers matches the expected value. Compare the attitude estimates to the absolute attitude from the Motion Tracking cameras.

The requirements for ADCS subsystem can be found in the Appendix 25.

8.1.2 Test Facilities, Setup and Configuration

T-AOC-01 requires the magnetorquer to be connected to a power source with the voltage set to 5 V. In order to measure the magnetic dipole produced by the magnetorquer the MATLAB Mobile Application was used. The mobile app utilises the magnetometer of the iPhone 14 Pro to record the magnetic flux density of the surrounding environment. The magnetorquer was connected to the power source with long wires so that the magnetorquer and phone could be placed as far away as possible from the power source to reduce any interference caused by the power source as seen in Figure 17. The phone was placed 0.30 meters away from the magnetorquer in line with the centre of the coil. The magnetic flux density sensing was then started taking samples at a frequency of 10 Hz. After roughly 30 seconds the power source was turned on and the magnetic flux density was measured for another 30 seconds.

To quantify the overall change in magnetic flux density due to the magnetorquer the mean magnetic flux density in each of the three directions over the first 30 seconds was calculated. A buffer of 3 seconds was then used to avoid any interference and then the mean magnetic flux density in each of



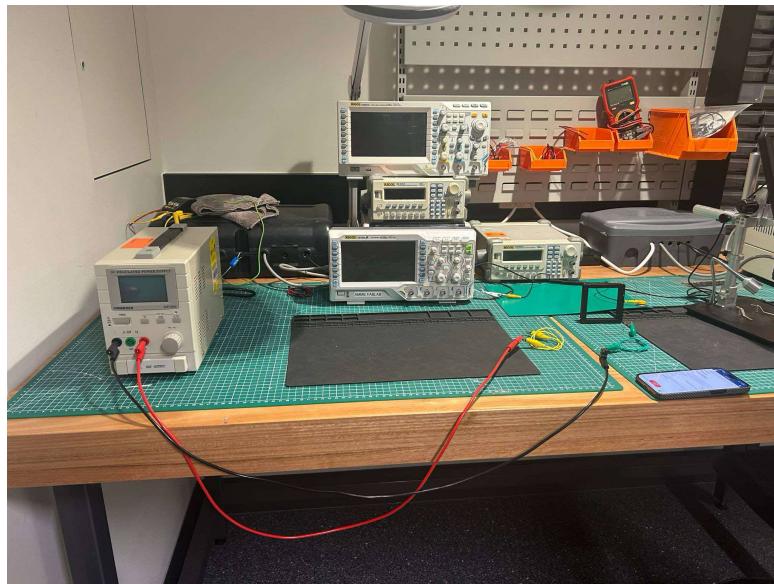


Figure 17: Magnetorquer testing setup

the three directions from 33 seconds to the final sample time was calculated. This essentially provided us with the magnetic flux density without and with the magnetorquer. The difference in each of the three directions was then calculated and these three components were normalised to obtain the overall change in magnetic flux density (ΔB). This was used with the following expression to calculate the magnetic dipole produced by the magnetorquer [2].

$$\vec{M}_{\text{dipole}} = \frac{4\pi}{\mu_0} \left[\frac{\frac{R_x}{L} - \frac{1}{2}}{\left(R_x^2 - R_x L + \frac{L^2}{4} \right)^{3/2}} - \frac{\frac{R_x}{L} + \frac{1}{2}}{\left(R_x^2 + R_x L + \frac{L^2}{4} \right)^{3/2}} \right]^{-1} \Delta B \quad (1)$$

Where R_x represents the distance between the magnetorquer and magnetometer (0.3 m), L represents the total width of all the coil (10×10^{-3} m for the flat magnetorquer) and μ_0 represents the permittivity of free space which is $4\pi \times 10^{-7}$ [2].

T-AOC-02 requires the sun sensor to be wired to the Raspberry Pi Pico on a breadboard. A phone torchlight will be used to shine light at varying angles.

T-AOC-03 and **T-AOC-04** requires the magnetorquers to be connected to H-Bridge through the ADCS PCB. A multimeter will be connected in series to measure the current output from the H-Bridge and the same mobile phone set up will be used to measure the change in magnetic flux density observed.

T-AOC-05 requires the IMU to be connected to power through a Raspberry Pi Pico on a breadboard.

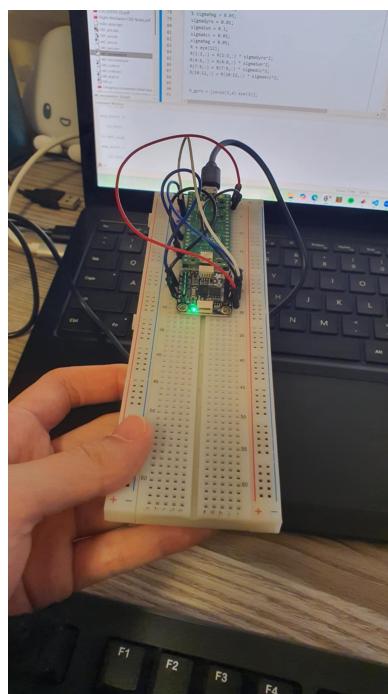


Figure 18: Picture of T-AOC-05 with the IMU Connected to the Raspberry Pi Pico on a BreadBoard

T-AOC-06 requires uploading the determination algorithm to the Raspberry Pi Pico on the ADCS PCB to ensure the conversion to C++ has not caused additional errors.

T-AOC-07 requires uploading the determination algorithm to the ADCS PCB which has the Raspberry Pi Pico and IMU connected.

T-AOC-08 and **T-AOC-09** required simulations to be performed in C++ using the actual attitude control functions in order to fulfill the testing requirements.

T-AOC-10 requires uploading the control algorithm to the Raspberry Pi Pico on the ADCS board, with the magnetorquers connected. A mobile phone is required to measure the magnetic flux.

T-AOC-11 requires uploading the determination and control algorithms to the Raspberry Pi Pico with filler sensor values.

T-AOC-12 requires uploading the determination and control algorithms to the ADCS PCB. The Air Bearing table and Motion Tracking cameras will be required to allow one axis freedom and absolute attitude.

8.1.3 Loads Adopted

T-AOC-01: the load being measured and tested was the magnetic flux density.

T-AOC-02: the load being measured and tested was the illuminance of the phone torchlight.

T-AOC-03, T-AOC-04: the load being measured and tested was the current flowing through the magnetorquers and the change in magnetic flux density.

T-AOC-05, T-AOC-06, T-AOC-07: the load being measured and tested in the sensor output from the IMU.

T-AOC-08, T-AOC-09: the load being measured and tested is the control torque values outputted from the control algorithms.

T-AOC-10: The load being measured and tested is the magnetic flux and the torque from the magnetorquers in the fully integrated system. The movement was visually observed and the

magnetic flux is measured with a mobile phone.

T-AOC-11, T-AOC-12: The Air Bearing table and Motion Tracking cameras will allow for the measurement and testing of the attitude estimates. In addition to being able to visually observing the torque generated by the magnetorquers.

8.1.4 Test Flow

This outlines the order in which the tests are completed:

1. **T-AOC-01, T-AOC-02, T-AOC-04 and T-AOC-05** are unit tests of critical components of ADCS and ensures they are operating as expected before they are soldered onto the ADCS PCB.
2. **T-AOC-03 and T-AOC-06** are unit tests of critical components once they have been soldered onto the ADCS board. This ensures no components were damaged during the soldering process.
3. **T-AOC-06, T-AOC-08 and T-AOC-09** are tests to ensure the determination and control algorithms are operating as expected. These tests use filler values to ensure the algorithms function before integrating with sensor values.
4. **T-AOC-07, T-AOC-010** are tests to ensure the determination and control algorithms work on the ADCS board and are able to interact properly with the IMU and the magnetorquers.
5. **T-AOC-11** ensures the full system operates as expected with simulated sensor measurements. This allows for a controlled environment to test and tune the algorithms.
6. **T-AOC-12** is the full integration of the ADCS system and will use the determination and control algorithms with the IMU sensor data and magnetorquer outputs.



8.2 Test Results

Table 18: ADCS Results

Test ID	Results	PASS/FAIL
T-AOC-01	A clear change in magnetic flux density was measured indicating a magnetic dipole moment sufficient to counteract the maximum external torque. This was observed for all three magnetorquers	PASS
T-AOC-02	There was a clear change in the illuminance measured, this corresponded to the angle at which the light was being shone. This was observed with four out of the five sun sensors. The fifth was damaged during soldering.	PARTIAL
T-AOC-03	The magnetometers were connected to the ADCS PCB and we obtained 1.296 A on the X axis magnetorquer, 1.301 A on the Y axis magnetorquer and 0.851 A on the Z-Axis Magnetorquer.	PASS
T-AOC-04	When the duty cycle on the H-Bridges were varied from 20% to 100% in increments of 20% a linear relationship was observed between the H-Bridge duty cycle and the magnetic dipole moment obtained using the magnetometer on a mobile phone.	PASS
T-AOC-05	The values observed were the quaternions, Euler angles and calibrated accelerometer, gyroscope and magnetometer values. These all changed corresponding to the axis the IMU was rotated around.	PASS
T-AOC-06	The filler sensor measurements were assumed to mimic the satellite rotating around the roll axis. The determination algorithm converged to this result within seconds.	PASS
T-AOC-07	The IMU was held still and also rotated around the roll axis, using the sensor measurements. The determination algorithm was able to converge within 20 seconds.	PASS
T-AOC-08	The simulated algorithm produced the correct magnetic dipole moment according to the B-Dot algorithm and calculated the correct duty-cycle when acting on sample attitude data.	PASS
T-AOC-09	The simulated algorithm produced the correct magnetic dipole moment according to the PID control law and calculated the correct duty-cycle when acting on sample attitude data.	PASS
T-AOC-10	Testing to be conducted on-day	
T-AOC-11	Testing to be conducted on-day	
T-AOC-12	Testing to be conducted on-day	

8.2.1 T-AOC-01

A clear change in magnetic flux density was measured indicating a magnetic dipole moment sufficient to counteract the maximum external torque. This was observed for all three magnetorquers as the dipole moment was larger than 0.016 Am. This can be seen in Table 19



Table 19: Maximum dipole moment produced by each of the magnetorquers.

Magnetorquer	Maximum Dipole Moment (Am)
X	0.0271
Y	0.0376
Z	0.1111

Result: PASS

8.2.2 T-AOC-02

The Sun Sensors were unit tested by determining whether they returned varying lux values for a phone torch passover. Out of the five photodiode units, four were successfully recognised by the PCB and returned results that varied in magnitude based on angle of incidence for a fixed source distance. The failure of one unit is not catastrophic, as the sun direction determination algorithm requires a minimum of three photodiodes.

The results of a single sensor chip with phone passover is given in Figure 19. The movement of the phone torch at a constant angular rate is reflected in the sinusoidal response of the lux values.

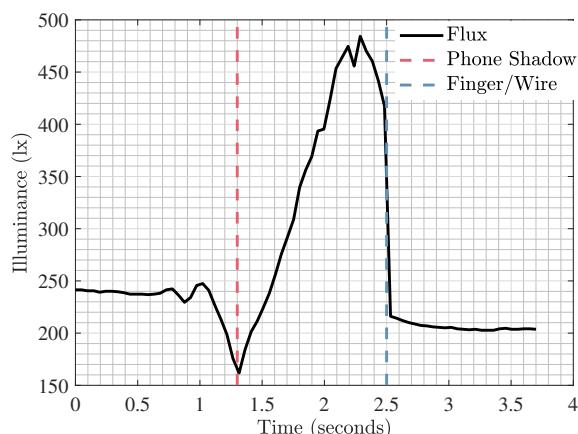
Result: PARTIAL

Figure 19: Sun sensor testing with phone light passover.

8.2.3 T-AOC-03

The magnetometers were connected to the ADCS PCB and we obtained 1.301 A on the X axis magnetorquer, 1.296 A on the Y axis magnetorquer and 0.851 A on the Z-Axis Magnetorquer. The testing setup can be seen in Figure 20 and Figure 21.



Figure 20: Current measured across the Y-Axis Magnetorquer using ADCS PCB



Figure 21: Current measured across the Z-Axis Magnetorquer using ADCS PCB

Result: PASS

8.2.4 T-AOC-04

When the duty cycle on the H-Bridges was varied from 20% to 100% in increments of 20%, a linear relationship was observed between the H-Bridge duty cycle and the magnetic dipole moment obtained using the magnetometer on a mobile phone. As seen in Figure 22, Figure 23 and Figure 24 all three magnetorquers can comfortably produce more than 0.016 Am of Dipole Moment which was that produced by the maximum external torque in the critical design review.

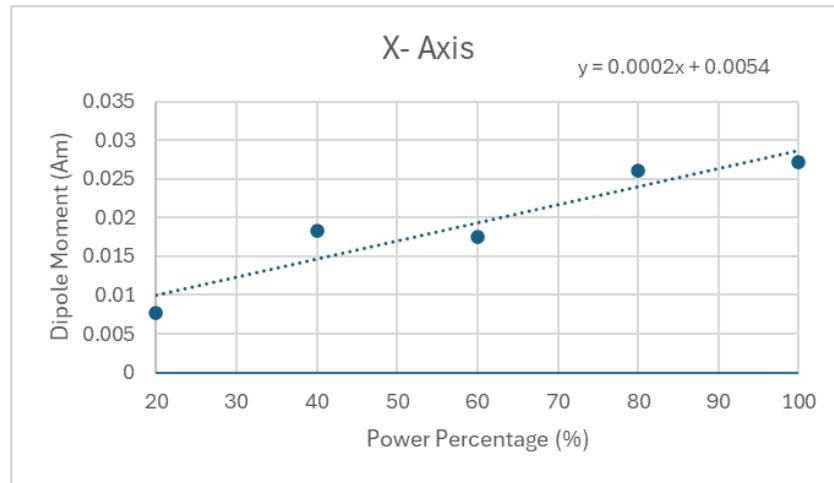


Figure 22: X-Axis Magnetorquer across duty cycle

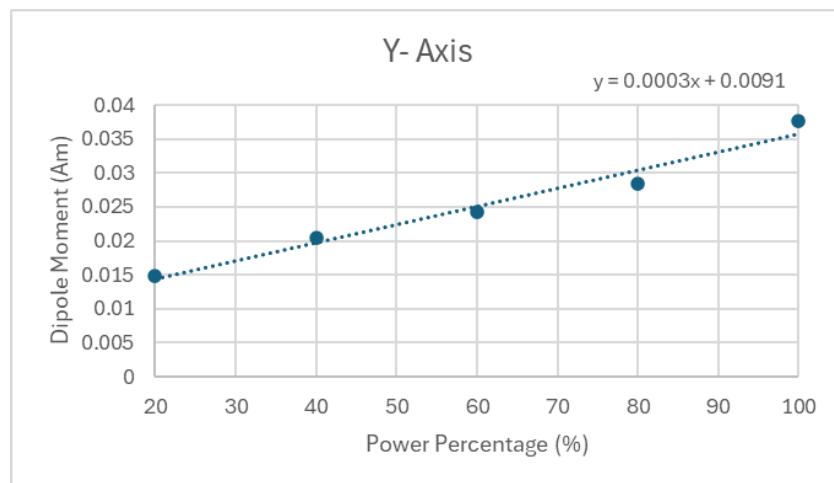


Figure 23: Y-Axis Magnetorquer across duty cycle

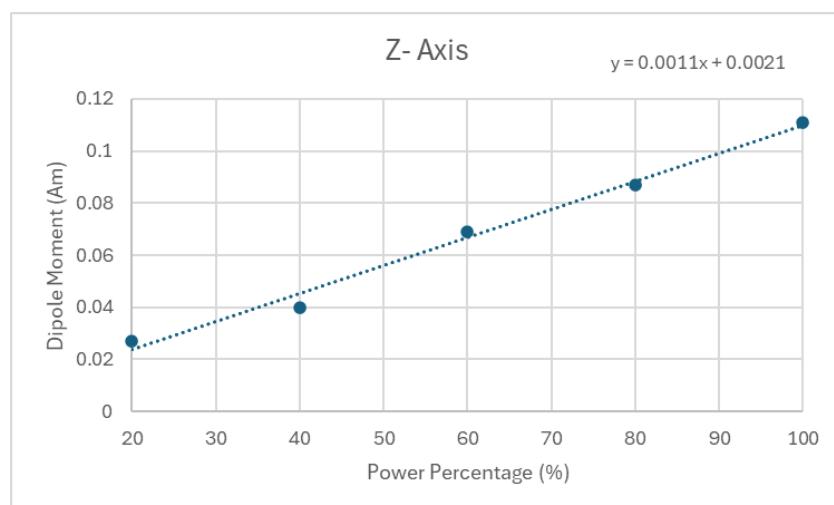


Figure 24: Z-Axis Magnetorquer across duty cycle

Result: PASS



8.2.5 T-AOC-05

The IMU connected to the Raspberry Pi Pico was able to output quaternions, Euler angles and calibrated sensor readings. These readings had minimal delay and was accurate in following the direction rotations about each axis.

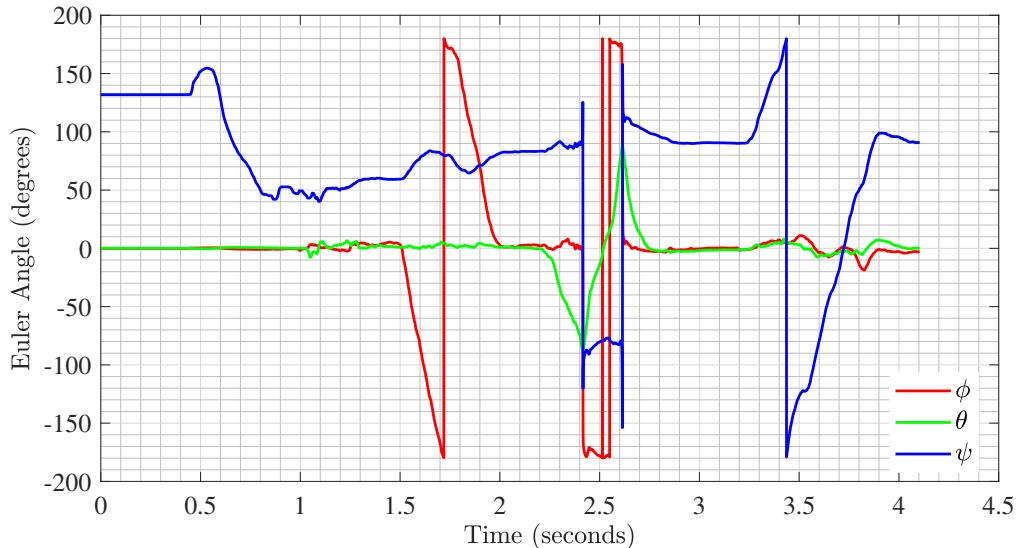


Figure 25: Euler Angles Obtained from BNO085 IMU Test

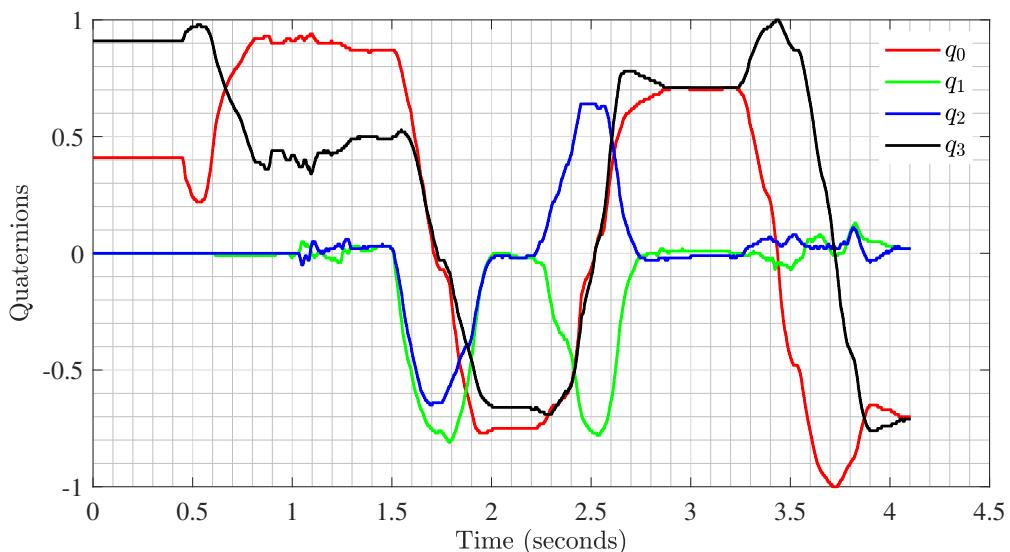


Figure 26: Quaternions Obtained from BNO085 IMU Test

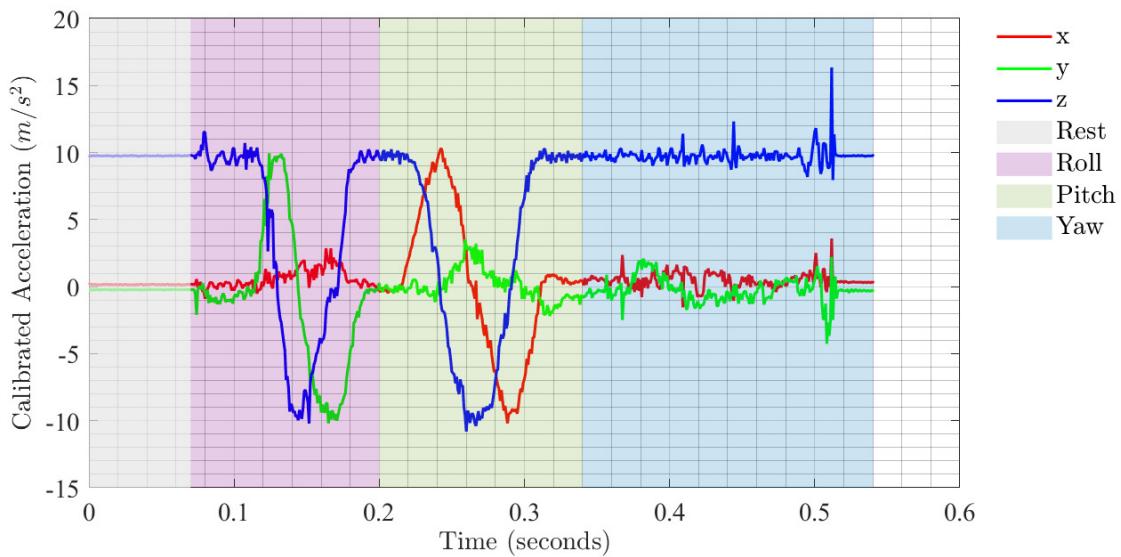


Figure 27: Calibrated Acceleration Obtained from BNO085 IMU Test

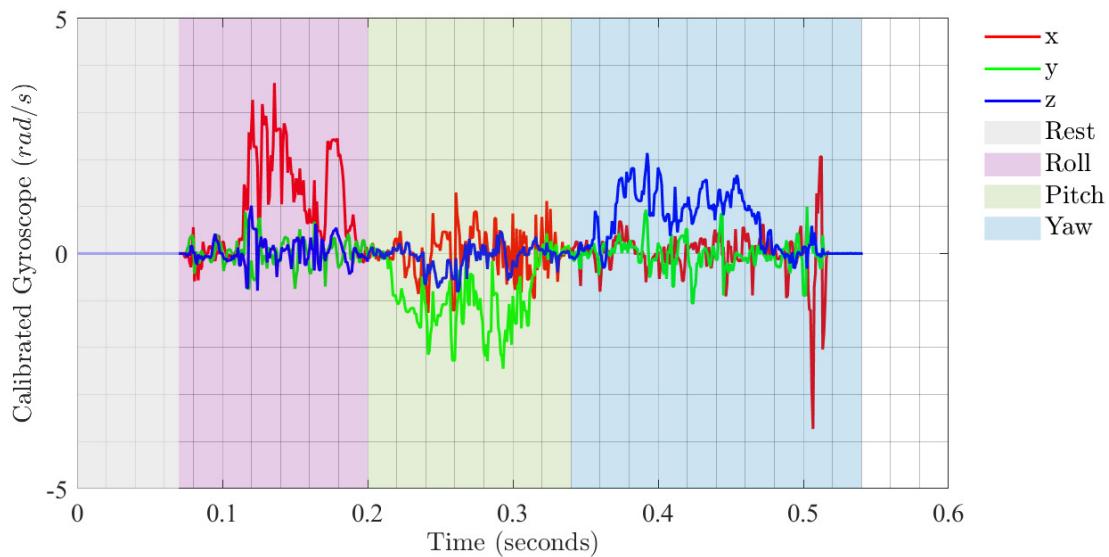


Figure 28: Calibrated Gyroscope Obtained from BNO085 IMU Test

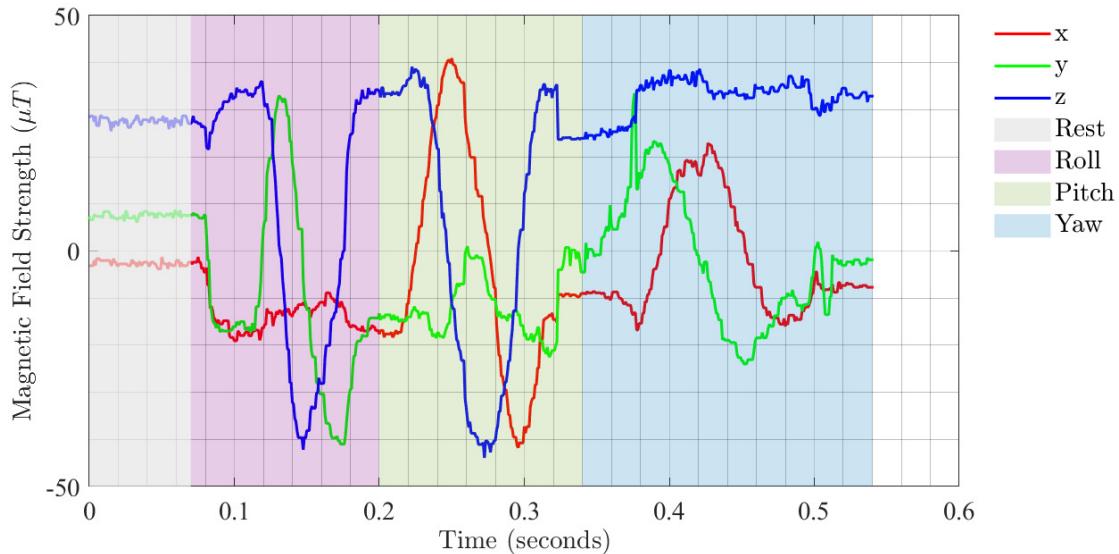


Figure 29: Magnetic Field Strength Obtained from BNO085 IMU Test

Result: PASS

8.2.6 T-AOC-06

The determination algorithm was tested through filler sensor values which simulated a 360° roll. It executed this with no errors and converged within seconds.

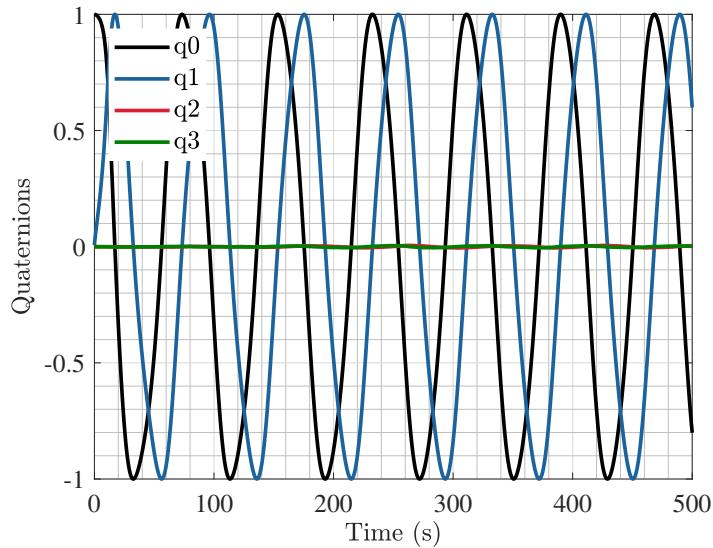


Figure 30: Estimated Quaternion Using Filler Sensor Measurements

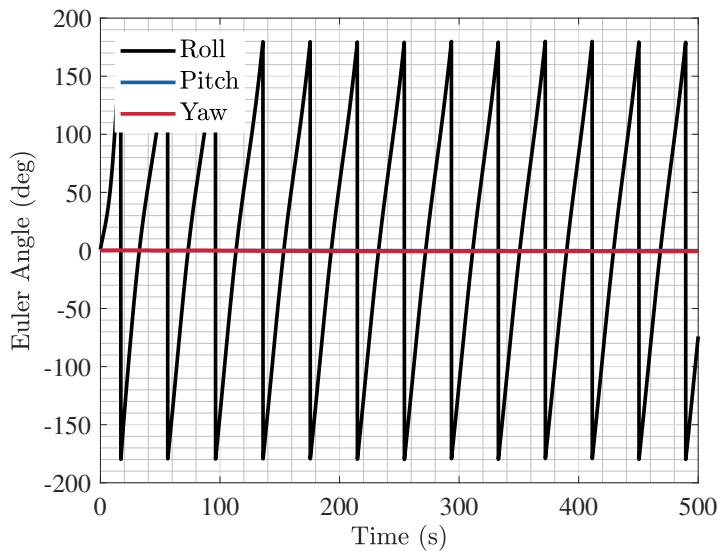


Figure 31: Estimated Euler Angles Using Filler Sensor Measurements

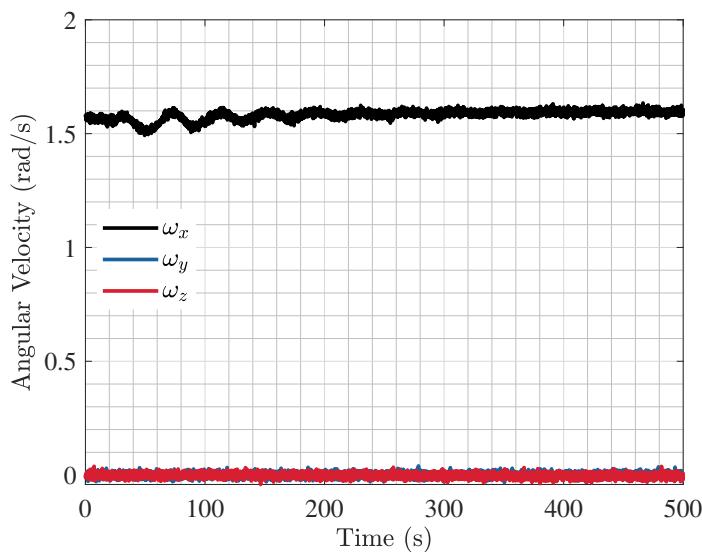


Figure 32: Estimated Angular Velocity Using Filler Sensor Measurements

Result: PASS

8.2.7 T-AOC-07

The determination algorithms were able to track the IMU when it was left to sit idle and during a roll. It took the algorithm approximately 20 seconds to converge. Thus more tuning and adjusting is required for it to perform without noticeable latency.

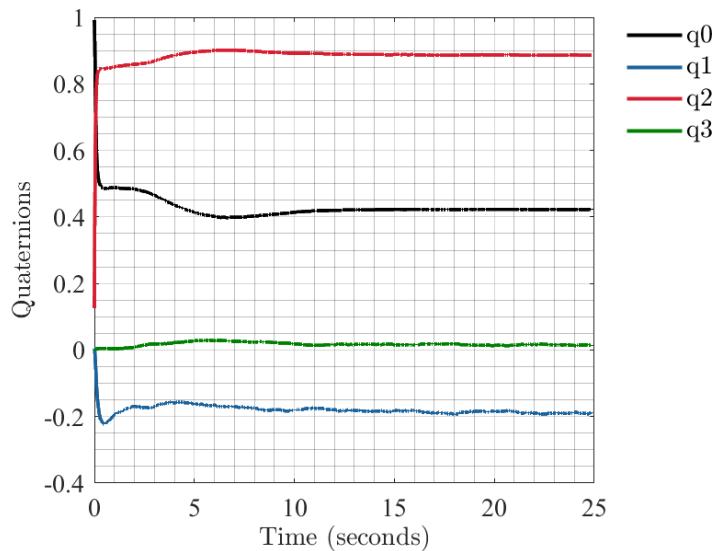


Figure 33: Estimated Quaternions with Sensor Values When IMU is Sitting Idle

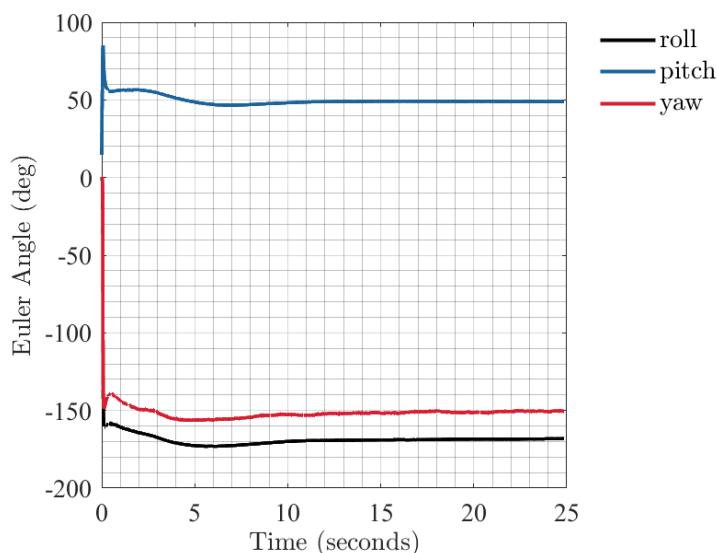


Figure 34: Estimated Euler Angles with Sensor Values When IMU is Sitting Idle

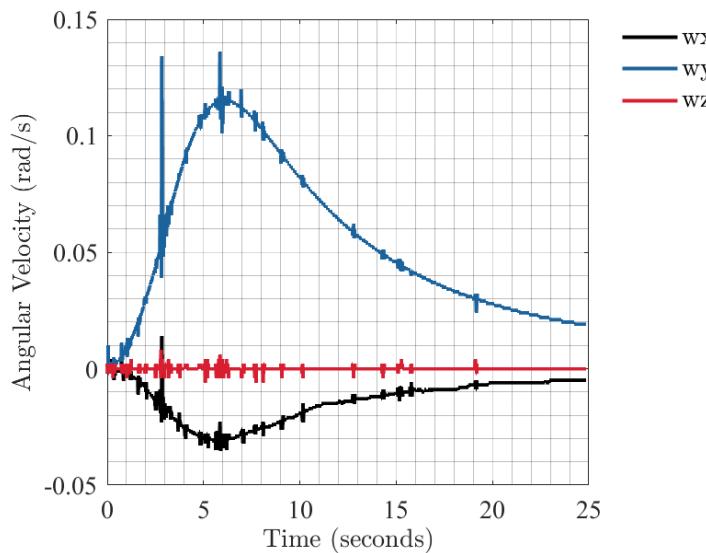


Figure 35: Estimated Angular Velocity with Sensor Values When IMU is Sitting Idle

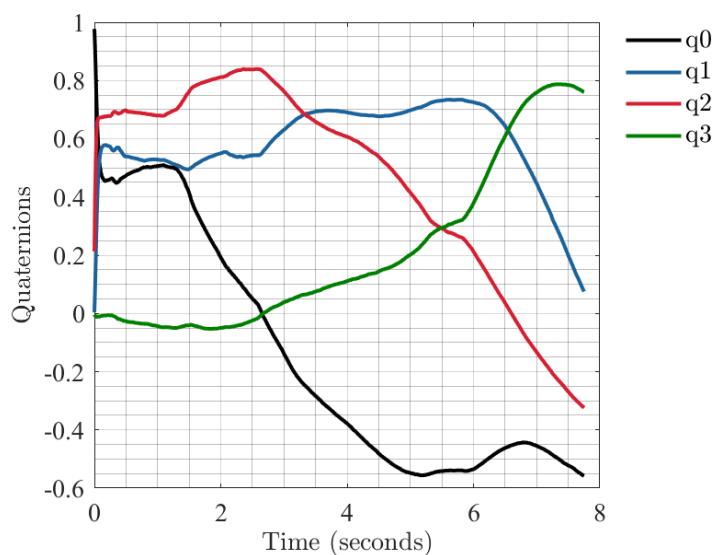


Figure 36: Estimated Quaternions with Sensor Values When IMU is Conducting a 360° Roll

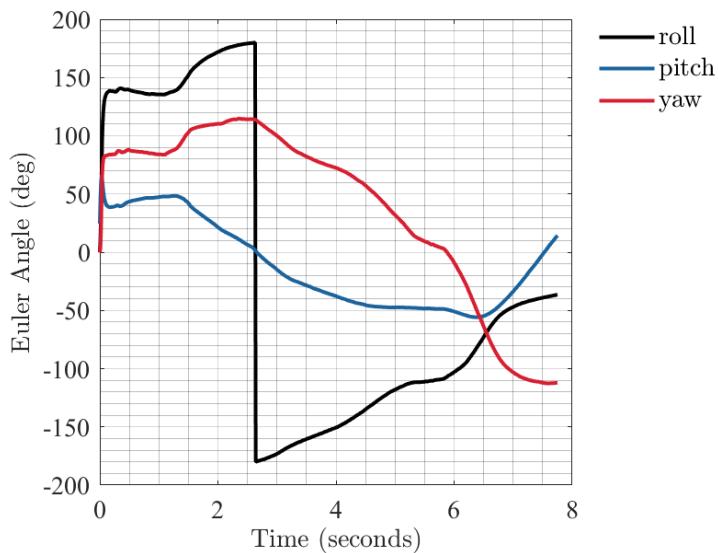


Figure 37: Estimated Euler Angles with Sensor Values When IMU is Conducting a 360° Roll

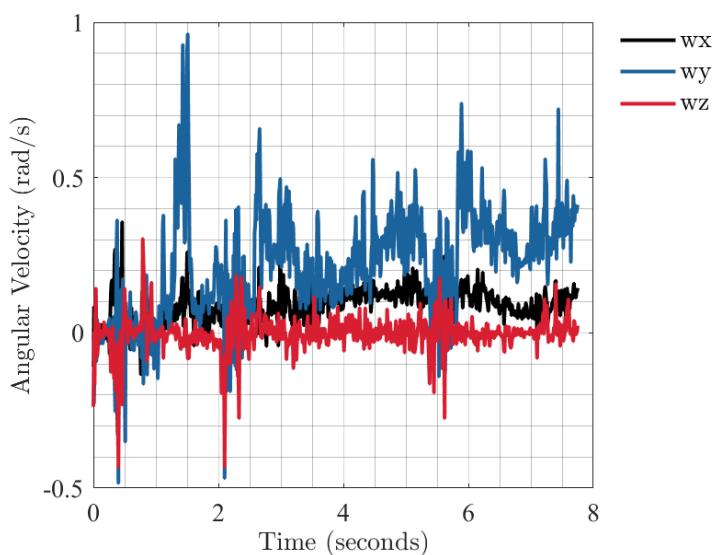


Figure 38: Estimated Angular Velocity with Sensor Values When IMU is Conducting a 360° Roll

Result: PARTIAL

8.2.8 T-AOC-08

The algorithm produced the correct direction of magnetic dipole moment and correct corresponding duty-cycle values for the H-bridge based on sample attitude coordinates. The algorithm was run from the command line in C++ using the functions that will be uploaded to the micro-controller on the ADCS - the outputs are shown in Figure 39.

```

==== Sample Satellite Data ====
Quat : [0.96 0.1 0.1 0.2 ]
Reference quat : [1 0 0 0]
Angular velocity (rad/s): [ 0.02 -0.01 0.015]
Body mag field (T) : [ 2.0e-05 1.5e-05 -2.5e-05]

==== B-dot Controller Test ====
Magnetic dipole moment (Am2) : [0.000125 0.004 0.0025 ]
PWM duty cycles (%) : [50.03 51. 50.62]

==== PID Controller Test ====
Magnetic dipole moment (Am2) : [0.0201981 0.0204981 0.0406462]
PWM duty cycles (%) : [55.05 55.12 60.16]
minijo@minijo-XPS-15-9530:~$ []

```

Figure 39: Testing of attitude control algorithm software

Result: PASS

8.2.9 T-AOC-09

Similarly to **T-AOC-08**, the algorithm produces the correct duty-cycle values for control of the H-Bridges based on sample attitudes coordinates. The results were shown in Figure 39.

Result: PASS

8.3 On-day Testing

T-AOC-10, **T-AOC-11** and **T-AOC-12** will be demonstrated on the formalised testing day. It will be conducted using the Air Bearing table and the Motion Tracking cameras. The expected results is the CubeSat will be able to provide accurate attitude estimates and the control algorithms will be able to actuate the magnetorquers to point the satellite in a specified direction. Some potential failure points may be the latency of the determination algorithm, overshoot of the control algorithm and overdraw of current from the H-Bridge. The way to fix this is to optimise our algorithms, tune our gains more carefully, and limit the duty-cycle of the PWM signal.



9 Structure

9.1 Test Definition

9.1.1 Requirements

Table 20: Structures Testing

Test ID	Req.	Description	Method	Pass Criteria
T-STR-01	S-STR-05	Verify hold-down release mechanism activates successfully	Hold down both switches and release; observe if the system powers on	The system shall activate when the switches are released
T-STR-02	S-STR-05	Verify deployment switch activates during release	Simulate deployment by actuating the release system and measure continuity across switch terminals	Switch signal shall transition appropriately with no premature deployment
T-STR-03	S-STR-04	Verify center of mass is within 20 mm of geometric center	SolidWorks mass property analysis	Center of mass must be within 20 mm in all directions
T-STR-04	S-STR-03	Verify total mass is under 2.66 kg	Weighing on calibrated scale and CAD mass calculation	Mass must be less than 2.66 kg
T-STR-05	S-STR-01	Verify envelope dimensions do not exceed $100\pm0.1 \times 100\pm0.1 \times 227\pm0.1$ mm	Manual measurement using calipers and ruler	All external dimensions within allowable envelope
T-STR-06	S-STR-06	Verify CubeSat frame does not cold-weld with deployer	Confirm all exposed aluminium contact surfaces are anodised	All exposed aluminium parts must be anodised to prevent cold welding

The requirements for structures subsystem can be found in the Appendix 24.

9.1.2 Test Facilities, Setup and Configuration

All structural tests were conducted in the the EXPLORER SPACE and FABLAB at the university. Measurement equipment included digital calipers, a calibrated laboratory scale, and test benches with rigid mounting fixtures. The SolidWorks 3D model of the CubeSat was used to extract mass properties, including total mass and center of gravity. The frame underwent surface treatment at an external certified anodising facility.

9.1.3 Loads Adopted

Structural and mechanical requirements were derived from CubeSat Design Specification Rev. 13. The structure was evaluated against expected mechanical loads through finite element analysis (FEA) and physical verification.



9.1.4 Test Flow

The order the tests are conducted is as follows:

1. **T-STR-01:** Verify switch operation via manual activation and multimeter reading.
2. **T-STR-02:** Simulate hold-down release event.
3. **T-STR-03:** Evaluate center of mass using SolidWorks mass properties tool.
4. **T-STR-04:** Weigh the CubeSat using a calibrated electronic scale.
5. **T-STR-05:** Measure external dimensions with ruler and calipers.
6. **T-STR-06:** Inspect anodising on external structural components in contact with deployer.

9.2 Test Results

Table 21: Structures Results

Test ID	Results	PASS/FAIL
T-STR-01	The hold-down release mechanism was successfully verified. When the two switches were held down and released, the onboard system powered up as expected.	PASS
T-STR-02	Deployment switch was triggered reliably during simulated release, and continuity confirmed correct timing with no premature actuation.	PASS
T-STR-03	SolidWorks analysis indicated the center of mass was within 13 mm of the geometric center in all directions.	PASS
T-STR-04	SolidWorks CAD estimated total mass at 2.043588kg. A calibrated scale confirmed the physical mass at 1.34 kg.	PASS
T-STR-05	Manual measurements showed structure dimensions were 100 mm x 100 mm x 226.5 mm, within the allowable tolerance.	PASS
T-STR-06	All exposed aluminium surfaces in contact with the deployer were anodised. Visual inspection confirmed uniform anodising coverage.	PASS

9.2.1 T-STR-01

The hold-down release switches were manually depressed and released while monitoring system power state. Upon release, the CubeSat powered on correctly, indicating that the hold-down release mechanism functioned as required.

Result: PASS

9.2.2 T-STR-02

A deployment simulation was performed by manually releasing the hold-down pins. Continuity checks across the deployment switch terminals confirmed correct signal transition. No premature deployment signals were detected.

Result: PASS

9.2.3 T-STR-03

The CubeSat 3D model was analyzed using SolidWorks. The calculated center of mass was 13 mm from the geometric center in each principal axis, meeting the requirement for balanced deployment dynamics.

Result: PASS



9.2.4 T-STR-04

Mass properties were extracted from the SolidWorks model and verified by weighing the assembled CubeSat. The digital scale measurement (1.34 kg) was below the modeled mass (2.043588kg), remaining under the 2.66 kg CubeSat mass limit.

Result: PASS

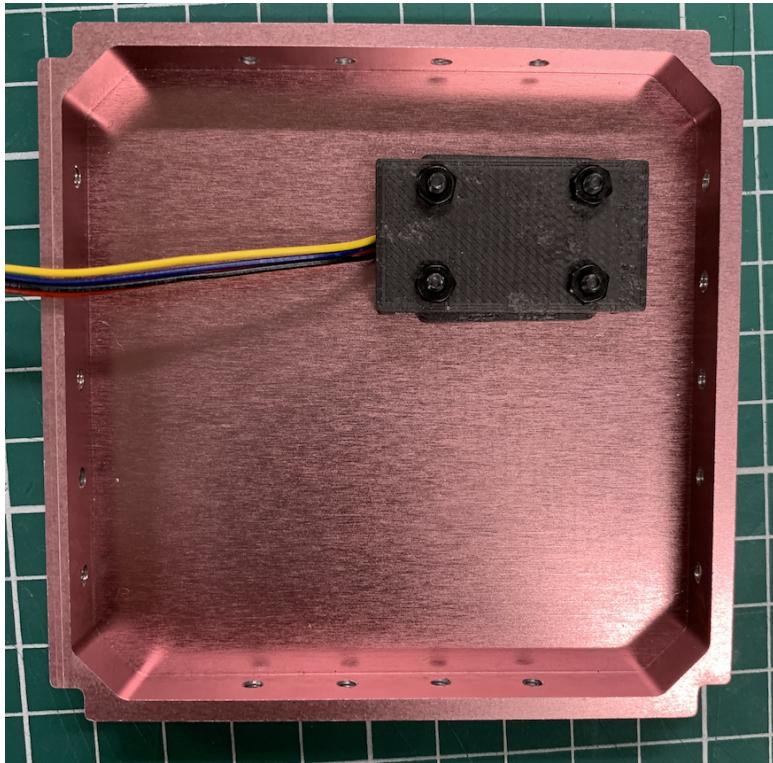
9.2.5 T-STR-05

Using calipers and a precision ruler, the CubeSat's external dimensions were measured. All dimensions fell within the 100 mm x 100 mm x 227 mm maximum envelope required by the CubeSat standard.

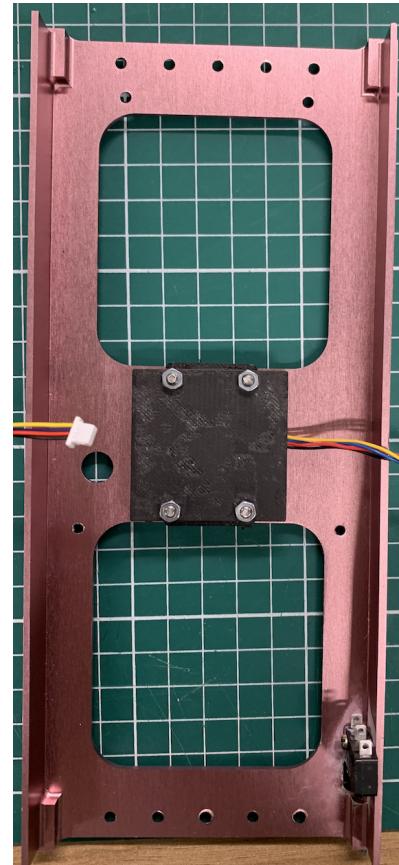
Result: PASS

9.2.6 T-STR-06

To comply with the cold-weld prevention requirement, all aluminium surfaces in contact with the deployer and associated hardware were anodised. Visual inspection confirmed continuous anodising coverage, and documentation from the treatment facility verified process compliance.



(a) Anodised end panel



(b) Anodised side panel

Figure 40: Anodised components of satellite (with pink shade)

Result: PASS

10 Payload

This section outlines the testing that was done on the payload of LUNATICS-0. This testing was done to ensure that the science objective could be completed successfully.

10.1 Test Definition

10.1.1 Requirements

Table 22: Payload Testing

Test ID	Req.	Description	Method	Pass Criteria
T-PAY-01	P-PAY-01	Spectrometer powers on.	Apply a 3.3 V voltage to the spectrometer power input pins, read from I2C port on spectrometer.	Values are received from the I2C ports.
T-PAY-02	P-PAY-01	Spectrometer produces accurate intensity curve for sources with unique emissions.	Expose spectrometer to a light source with known spectrum	Experimental curve matches known spectrum and an element/- colour can be identified.
T-PAY-03	P-PAY-03	The spectrometer returns continuous data	Read data from I2C and ensure updates at regular intervals.	A continuously-changing spectrum can be visualised, representing real-time sensor data.
T-PAY-04	P-PAY-01 P-PAY-03	The spectrometer only takes measurements when the satellite is set to PAYLOAD.	Initiate PAYLOAD mode from OBC.	Sensor readings are only received and stored when satellite is in PAYLOAD mode.

The requirements for payload subsystem can be found in the Appendix 31.

10.1.2 Test Facilities, Setup and Configuration

The tests for the payload subsystem were conducted in a controlled laboratory environment, assessing the individual component and subsystem both pre- and post-assembly.

The payload was tested to ensure the function of both the hardware and software, as well as the scientific feasibility to achieve the mission goal. These could all be demonstrated through a similar test of shining different coloured light onto the sensor, transmitting that data to the ground station, and then plotting both an intensity and absorption spectrum.

10.1.3 Loads Adopted

No physical loads were applied to the payload during testing.

All tests required data to be sent to the OBC via I2C, and then transmitted to the ground station over WiFi. Each test also required light to be incident on the sensor such that data could be returned and read.



10.1.4 Test Flow

T-PAY-01: is a unit test of the payload sensor component, to ensure that power is supplied to the unit. A 3.3V voltage and ground was connected to the payload.

T-PAY-02: Coloured filters were applied to the front of the CubeSat such that the ambient light passed through before reaching the sensor. The filters will restrict the wavelengths which can pass through, simulating the absorption of certain wavelengths by fake elements Redium (red filter), Yellowine (yellow filter), Greenium (green filter), Blueogen (blue filter), and Purpleium (purple filter). Therefore, by observing the resulting spectrum, we are able to determine the composition of the “atmosphere” by identifying the missing wavelengths or the resulting spectra.

T-PAY-03: The OBC is turned on and state machine begins running. The OBC is put in **NOMINAL** state, which makes it take payload readings, sending it to the OBC over I2C, and then saving it on the memory of the OBC. After 1 minute, change the state to **PAYOUTLOAD** and downlink the data.

T-PAY-04: The OBC will be set in **NOMINAL** state. The output will be observed, and it should be none. Then, the OBC will be set to **PAYOUTLOAD** and the resulting spectra observed.

10.2 Test Results

Table 23: Payload Results

Test ID	Result	PASS/FAIL
T-PAY-01	Power was successfully supplied to the spectrometer, demonstrated by the power light turning on on the sensor.	PASS
T-PAY-02	The wavelength intensities across the spectrum changed when exposed to light of different composition. A neutral observer was able to correctly identify the colours exposed to the payload without knowing what they were.	PASS
T-PAY-03	Data was transmitted from the payload through I2C to the ground station and displayed with little delay as a spectrum.	PASS
T-PAY-04	No recordings are taken from the sensor unless the satellite is in PAYOUTLOAD mode. During this time, the payload is still ‘on’ but draws minimal power.	PASS

10.2.1 T-PAY-01

Power was successfully supplied to the spectrometer. The red visible/infrared-region source LED is turned on in Figure 41.



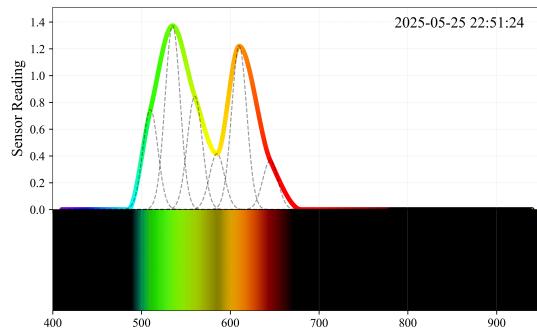
Figure 41: Payload testing showing red LED (power) on

Result: PASS

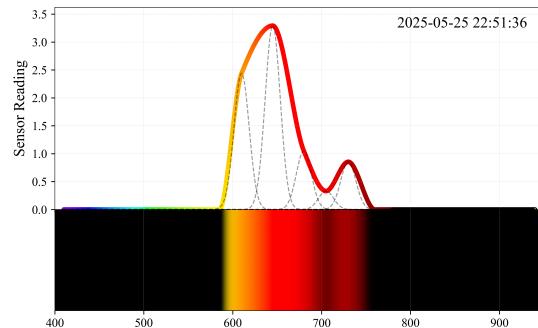
10.2.2 T-PAY-02

The spectrometer, in lieu of it not being of sufficient quality to have a resolution high enough to distinguish specific wavelengths, was alternatively able to distinguish between different coloured filters. Figure 42 demonstrates the resulting intensity and absorption spectra for a green- and red-filtered light, i.e., in the mock model, that the light passed through Greenium and Redium.

Result: PASS



(a) Green filter.



(b) Red filter.

Figure 42: Unique emission spectra for different coloured filters (atmospheric compositions).

10.2.3 T-PAY-03

The satellite, when set to PAYLOAD mode, transmitted live readings from the payload spectrography sensor, which were then displayed live (or with a very short delay) on the ground station GUI. A real-time demonstration can be seen in the demonstration video. A screenshot from the video can be seen in Figure 43, where the blue filter simulates the absorption of wavelengths by Blueogen.

Result: PASS



(a) Blue filter over payload.



(b) Resulting measured spectra.

Figure 43: Results from live demonstration of payload.

10.2.4 T-PAY-04

The ground station interface displays a continuously-changing intensity (top) and absorption spectrum depending on the incoming light only when set to PAYLOAD mode. Otherwise, when not in this mode, no measurements are taken and the GUI displays a message relaying as such (see Figure 44).

Result: PASS



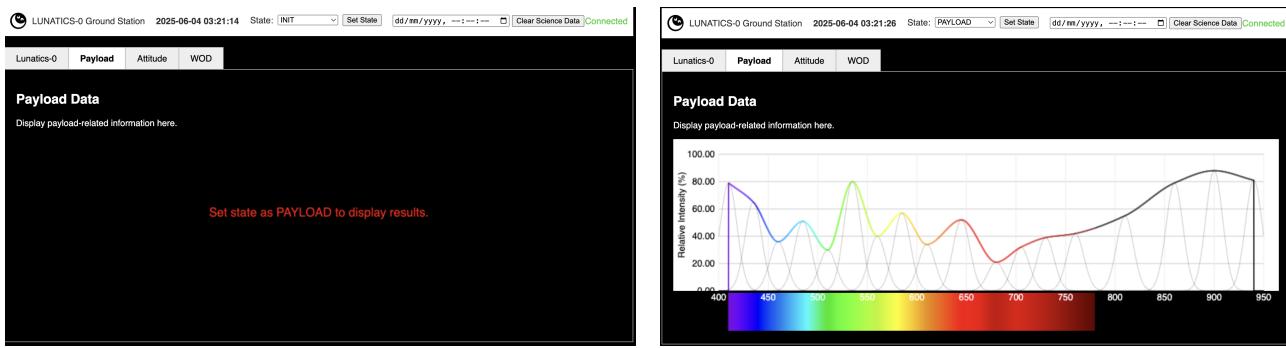


Figure 44: GUI depending on whether satellite is in or not in PAYLOAD state.

References

- [1] P. Fortescue, M. Swartwout, and P. Dervan, "Thermal design of cubesat payloads: A review of lessons learned and future challenges," *Acta Astronautica*, vol. 115, pp. 247–256, 2015. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0094576515001940>
- [2] J. Amin, "The design, assembly, and testing of magnetorquers for a 1u cubesat mission," Georgia Institute of Technology, Atlanta, GA, AE 8900 MS Special Problems Report, December 2019, advisor: Dr. E. Glenn Lightsey. [Online]. Available: <https://ssdl1.gatech.edu/sites/default/files/ssdl-files/papers/mastersProjects/AminJ-8900.pdf>
- [3] X. Wu, "AERO4701 USyd CubeSat System Requirements and Recommendations," University of Sydney, Tech. Rep., February 2025.



Appendix A Mission requirements

The following Mission Requirements were taken, adapted, and expanded upon from the University of Sydney AERO4701 CubeSat System Requirements and Recommendations [3].

A.1 CubeSat System Requirements

Table 24: Structural system requirements.

Requirement ID	Description	Verification Criteria	Verification Method	Phase
S-STR-01	The CubeSat SHALL fit within a $100 \times 100 \pm 0.1$ mm footprint.	Measurement within dimension	Dimension measurement	AIT
	The CubeSat SHALL have a height of 227.0 ± 0.1 mm.	Measurement within dimension	Dimension measurement	AIT
	The CubeSat SHALL have feet of $8.5 \times 8.5 \pm 0.1$ mm.	Measurement within dimension	Dimension measurement	AIT
	The CubeSat's external edges SHALL be rounded $R \times 1$ mm or chamfered $45^\circ \times 1$ mm.	Measurement within dimension	Dimension measurement	AIT
S-STR-02	The CubeSat SHALL use the reference frame as given.	Design stage	ADCS testing	PDR
S-STR-03	The CubeSat SHALL not exceed a mass of 2.66 kg.	Mass ≤ 2.66 kg	Mass measurement	AIT
S-STR-04	The CubeSat SHALL have its COG within 10 mm of the geometric centre.	COG within limit	COG test	AIT
S-STR-05	The deployment switches SHALL be non-latching (electrically or mechanically).	Deployment switches do not latch	Design	PDR
S-STR-06	The rails and standoffs SHALL be constructed of material that cannot be cold-welded to adjacent materials.	Design stage	Materials testing	CDR

Table 25: Attitude & Orbit Control Subsystem (AOCS) requirements.

Requirement ID	Description	Verification Criteria	Verification Method	Phase
S-AOC-01	The CubeSat SHALL be able to recover from tip-off rates up to $\pm 50^\circ \text{ s}^{-1}$ within two days (nominal conditions).	Attitude recovered	Simulation	CDR
	The CubeSat SHOULD be able to recover from tip-off rates up to $\pm 90^\circ \text{ s}^{-1}$ in off-nominal situation.	Attitude recovered	Simulation	CDR
S-AOC-02	The CubeSat SHALL be able to point its direction within $\pm 5^\circ$.	Pointing within the tolerance	Simulation and measurement	CDR
	The CubeSat SHOULD be able to point its direction within $\pm 2^\circ$.	Pointing within the tolerance	Simulation and measurement	CDR
S-AOC-03	The CubeSat SHALL maintain its orbit for six months.	Simulated satellite remains in orbit	Simulation	AIT
	The CubeSat SHOULD maintain its orbit for twelve months.	Simulated satellite remains in orbit	Simulation	CDR

Table 26: Electrical Power System (EPS) requirements.

Requirement ID	Description	Verification Criteria	Verification Method	Phase
S-EPS-01	The power supply SHALL be provided at appropriate voltage by solar array or battery to meet power requirements for all subsystems in all modes of operation.	Satellite runs in all modes	EPS testing	AIT
S-EPS-02	The CubeSat SHALL be able to be commissioned in orbit following the last powered-down state without battery charging, inspection, or functional testing for a period of up to eight months.	Returns after powered down	EPS testing	AIT
S-EPS-03	The CubeSat SHALL be powered off during the launch until deployed from the deployment system.	Satellite is off when not deployed	EPS testing	AIT
S-EPS-04	The CubeSat SHALL have battery current surge and over-charge protection circuits in place.	Batteries do not receive excess current	EPS testing	AIT

Table 27: Onboard Computer & Data Handling (OBC/OBDH) requirements.

Requirement ID	Description	Verification Criteria	Verification Method	Phase
S-OBC-01	The CubeSat SHALL collect Whole Orbit Data (WOD) and log telemetry every minute for the entire duration of the mission. The WOD must contain: time; spacecraft mode; battery bus voltage; battery bus current; current on regulated 3.3V bus; current on regulated 5V bus; communication subsystem temperature; EPS temperature; and battery temperature.	Data passes software tests for data packaging	OBC Testing	CDR
S-OBC-02	The WOD SHALL be stored on the OBC until they are successfully downlinked.	Data before transmission is received	OBC Testing	AIT
S-OBC-03	Any computer clock used on the CubeSat and on the ground segment SHALL exclusively use Coordinated Universal Time (UTC) as time reference.	Computers use UTC	OBC Testing	CDR
S-OBC-04	The OBC SHALL have a real time clock information with an accuracy of 500 ms during science operation. Relative times should be counted/stored according to the epoch 01.01.2000 00:00:00 UTC.	Time stamps of data within tolerance	OBC Testing	CDR
S-OBC-05	The OBSW programmed and developed SHALL only contain code that is intended for use on that CubeSat on the ground and in orbit.	Software is necessary	Design	CDR
S-OBC-06	The OBSW code SHALL include comments and sensible variable names such that functionality and control flow behaviour can be understood.	Code conforms to Google C++ Style Guide	Design	CDR
S-OBC-07	A command SHALL be implemented that can delete any SU data held in mass memory originating prior to a Date-Time stamp given as a parameter.	Data deleted successfully	OBC Testing	AIT
S-OBC-08	The OBC SHALL restart if it stops functioning for 60 seconds.	OBC Restarts	OBC Testing	AIT
S-OBC-09	The software on the OBC SHOULD have deterministic execution	Code runs on a single thread	Design	CDR
S-OBC-10	The software on the OBC SHALL have backup paths and fail over methods	Code uses error management	Design	CDR
S-OBC-11	The software SHALL handle single-event upsets	Code checks the validity of the received data	Design	CDR

Table 28: Telemetry, Tracking & Command (TT&C) requirements.

Requirement ID	Description	Verification criteria	Verification method	Phase
S-TTC-01	The CubeSat SHALL have and make use of its unique satellite ID.	ID used in data transmission	COMMS testing	PDR
S-TTC-02	The CubeSat SHALL transmit the current values of the WOD parameters and its unique satellite ID through a beacon at least once every 30s. The beacon SHOULD be transmitted every 10s during LEOP phase to allow for multiple receptions of the beacon during a pass.	Packets received successfully	COMMS testing	AIT
	The CubeSat SHALL use the AX.25 Protocol.	Packets received successfully	COMMS testing	AIT
S-TTC-03	The data type during downlink SHALL be specified in the Secondary Station Identifier (SSID) in the destination address field of the AX.25 frame. Science data SHALL be indicated using 0b1111 and WOD with 0b1110	Data follows protocol Data type specified	COMMS testing	CDR
S-TTC-04	A user-friendly and documented software SHALL be implemented, consisting of a) CubeSat data Frames Decoder; b) CubeSat data Packet decoder; and c) CubeSat data Viewer that complies with radio amateur regulations.	Data indicated properly Readable, commented software	Design	CDR

Table 29: Thermal Control Subsystem Requirements

Requirement ID	Description	Verification Criteria	Verification Method	Phase
S-THE-01	The CubeSat SHALL maintain all its electronic components within their operating temperature range while in operation and within the survival temperature range at all other times after deployment.	Component temperatures within range	Thermal testing/simulation	CDR
S-THE-02	The CubeSat SHALL survive within the temperature range of -40°C to $+80^{\circ}\text{C}$ from the time of launch until its end of life.	Data transmission continues during thermal cycling	Thermal testing/simulation	CDR
S-THE-03	The CubeSat SHALL undergo Thermal Vacuum Cycling and Bake-Out testing to verify its thermal performance in a space-like environment.	Compliance with Thermal Vacuum requirements	Thermal testing/simulation	CDR
S-THE-04	The CubeSat SHALL not exceed a temperature variation rate of $\geq 1^{\circ}\text{C}/\text{min}$ during testing.	Measured thermal variation	Thermal testing/simulation	CDR

Table 30: General system requirements.

Requirement ID	Description	Verification criteria	Verification method	Phase
S-GEN-01	The CubeSat SHALL be designed to have an in-orbit lifetime of at least six months.	In-orbit lifetime \geq 6 months	Design	PDR
S-GEN-02	The CubeSat SHALL not use any material that has the potential to degrade in an ambient environment during storage after assembly (as long as two years).	Materials do not degrade within 2 years	Design	PDR
S-GEN-03	The CubeSat SHALL withstand total contamination of 3.1 mg/m^2 at all phases of the launch vehicle ground operation and in-flight.	Components withstand contamination	Design	CDR
S-GEN-04	“Apply Before Flight” items, including tags or labels, SHALL not protrude past the dimensional limits of the CubeSat when fully inserted.	No protrusions on CubeSat	Visual inspection	AIT
S-GEN-05	“Remove Before Flight” items SHALL be identified by a bright red label of at least 4 cm^2 containing the words “REMOVE BEFORE FLIGHT” and the name of the satellite printed in large white capital letters.	Clearly visible	Visual inspection	AIT
S-GEN-06	The CubeSat ID SHALL be printed, engraved, or marked on the CubeSat.	Clearly visible	Design	AIT

A.2 Payload Requirements

Table 31: Science and payload mission requirements.

Requirement ID	Description	Verification criteria	Verification method	Phase
SCIENCE				
P-SCI-01	Spectral data SHALL be gained across the entire atmosphere over the mission duration.	Measurements over defined grid.	Simulation	CDR
P-SCI-02	The CubeSat SHALL be in a sun-synchronous orbit.	Orbit is sun-synchronous	Simulation	PDR
P-SCI-03	The CubeSat SHALL maintain a Nadir-pointing orientation with an attitude stability of $\pm 5^\circ$.	Pointing vector has $\leq \pm 5^\circ$ variation.	ADCS testing	CDR
P-SCI-04	Spectral data SHALL be recorded only when reflected sunlight is available.	Incoming light intensity $>$ dark photocurrent.	Hardware testing	CDR
P-SCI-05	The mission SHALL have a lifetime of at least one year to achieve full global atmospheric mapping.	Mission lifetime > 1 year	Simulation	PDR
PAYOUT				
P-PAY-01	The spectrometer payload SHALL observe light over a visible to near-infrared wavelength range.	Return wavelengths across range.	Payload testing	PDR
P-PAY-02	The payload SHALL have a field of view that allows for a minimum swath width of 200 km.	Swath width ≥ 200 km.	Simulation	PDR
P-PAY-03	The payload SHALL take measurements at regular intervals with at least 90% overlap to ensure high resolution data readings.	Measurements taken at 2s+.	Design	CDR
P-PAY-04	The CubeSat SHALL downlink at least 2 MB per pass using UHF/S-band with ground station access at least once per day.	2 MB transmitted within visible time frame	COMMS testing	AIT

Appendix B Code

B.1 OBC Code

<https://github.com/James-h-1969/lunatics-obc>

B.2 ADCS Code

<https://github.com/James-h-1969/lunatics-adcs>

B.3 Ground Station Code

<https://github.com/James-h-1969/space-ground-station>

Appendix C AIT Video

CLICK HERE to link to the AIT video.

https://unisydneyedu-my.sharepoint.com/:v/g/personal/elev3158_uni_sydney_edu_au/Eem1WwvtatM1WNV2cyuaNYBHR6U2aBMmY4waRBExodn0w?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJPbmVEcm12ZUZvckJ1c2luZXNzIiwicmVmZXJyYWxBcHBQbGF0Zme=0SHIZn

