Rotator System Test Plan

Mobile Antenna Tracking System (MATS)

9/26/2025

Zachary Murray, Dylan Vilcek, Ma Minghao

This document outlines the subsystem verification and test plan report for the Rotator subsystem aboard the MATS. This document will outline the objectives of the test, equipment and software required, as well as the procedures to verify the proper operation of the subsystem, and the results that were obtained during the testing of this system.

Table of Contents

[Executive Summary 3](#_Toc193873105)

[Introduction 4](#_Toc193873106)

[System Requirements and Specifications 4](#_Toc193873107)

[Subsystem Requirements and Specifications 5](#_Toc193873108)

[Objectives 6](#_Toc193873109)

[Required Equipment 6](#_Toc193873110)

[Testing Procedure 6](#_Toc193873111)

[System Self-Test 7](#_Toc193873112)

[Azimuth Mobility Test 9](#_Toc193873113)

[Elevation Mobility Test 10](#_Toc193873114)

[Command Response Verification Test 11](#_Toc193873115)

[Rotational Mobility Test 13](#_Toc193873116)

[Power Consumption Test 15](#_Toc193873117)

[Conclusion 16](#_Toc193873118)

[Appendix 17](#_Toc193873119)

## Executive Summary

The testing for the rotator proved to have a few challenges. Primarily, it was discovered that the library that is provided for the servo motors does not successfully compile and function on ESP32. Customer support has been contacted to resolve the problem but has not been as of time of testing. To address this, the part was revised to utilize an Arduino Mega R3 Development board. The only difference between these two boards is that the Mega R3 requires an input voltage of 7-12V. The testing was performed as outlined in the steps, and the accompanying tables have been filled out with the values which have been obtained. With the revised part, the Rotor was found to perform well within specification on functional motor movements. Power consumption was low, but we expect that in the future, with a more appropriate load on the rotator that the power consumption will greatly increase. A note on I2C, as this was originally in specification, the test was performed without I2C, because of the board swap we did not have development time to integrate I2C, as significant time was spent debugging the ESP32 library. The test plan has been revised to utilize this testing code, and I2C will be further integrated later.

## Introduction

The purpose of this subsystem verification plan is to establish a structured and repeatable process for verifying the performance, functionality, and compliance of the Rotator subsystem. This document provides a detailed framework for conducting verification activities, ensuring that all requirements are met before integration into the larger MATS. By following this plan, verification activities will be performed efficiently, ensuring the subsystem meets its design and functional requirements before proceeding to the next phase of system development.

## System Requirements and Specifications

The MATS aims to be small, lightweight, and cost effective, providing an encompassing solution compared to other commercial systems, which are large, heavy, and expensive. The MATS encompasses all the hardware required to accurately position the antenna, according to the TLE information for a given satellite. Utilizing GPS for positional data, the only input the user requires is to know which satellite is passing overhead. The MATS will be deployed by supplying power from any power system onboard many aircraft, marine vessels, emergency vehicles, and power systems commonly found in developing communities. Figure 1 shows the overall system diagram, showing how the subsystems interact with each other. The requirements enable the MATS to be low cost and reliable, while still having accurate satellite tracking.

A diagram of a power supply system

AI-generated content may be incorrect.

Figure 1: System Diagram

## Subsystem Requirements and Specifications

The Rotator system is responsible for rotating the antenna, moving it to a position given by the receiver system to track low earth orbit satellites. Due to the requirements given by the receiver, this system needs to receive an I2C signal for obtaining the azimuth and elevation location required and positioning the antenna to the specific coordinate within the specific time. System requirements can be found in Table 1.

Table 1: Rotator Subsystem Requirements

|  |  |
| --- | --- |
| **Requirement** | **Expectation** |
| Power Requirement | 3V3, 12VDC |
| Interface | I2C |
| Azimuth Accuracy | ±1.6° |
| Elevation Accuracy | ±1.6° |
| Power Consumption | < 75W |
| Cost | < $110 |

A diagram is provided in Figure 2 outlining the functional blocks and connections within the Rotator.

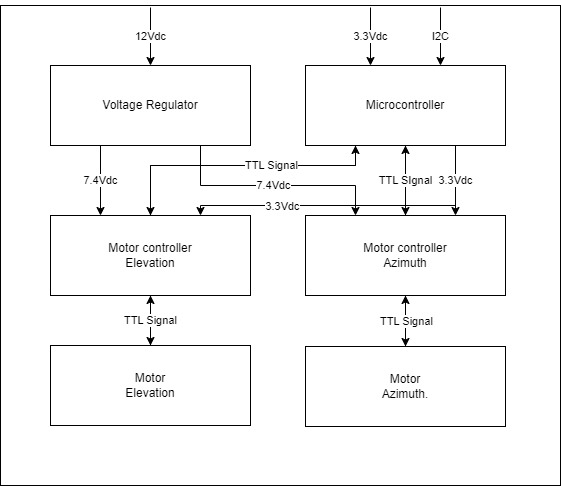


Figure 2: Rotator Block Diagram

These requirements ensure the proper operation of the Rotator, allowing the MATS to accurately track low earth orbit satellites.

## Objectives

These tests are intended to test the Rotator for proper operation. The rest of this document outlines the equipment and tests required to confirm that the Rotator is working as intended, and ready to be integrated into the larger MATS system. In these tests, the functional tests include:

* System Self-Test
* Azimuth Mobility Test
* Elevation Mobility Test
* Command Response Verification Test
* Rotational Mobility Test
* Power Consumption Test

## Required Equipment

For the following tests to be performed, some equipment should be obtained. This equipment is chosen for its simplicity of use and being readily available. The equipment required to perform these tests include:

* Laptop
* DMM
* Power Supply
* I2C Generator (raspberry pi, microcontroller, analog discovery)
* Stopwatch
* Protractor
* Wattmeter
* Compass

## Testing Procedure

This section details the verification procedures required to validate the Rotator. Each test ensures compliance with functional, performance, and reliability requirements. Data acquired from these tests should be documented in the provided tables.

### System Self-Test

The system self-test’s purpose is to verify that the azimuth and elevation motors start, perform the respective self-checks, and return to the original position. Table 2 outlines the procedure and measurements to be taken.

Table 2: System Self-Test Procedure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect power supply to Subsystem Under Test (SUT) as shown in Figure 3. | N/A |  |  |
| 2 | Using a protractor, take an initial position measurement at 90 degrees. | 90 |  |  |
| 3 | Turn on the power to the SUT and observe the behavior. | N/A |  |  |
| 4 | Verify the servo motors have performed their self-start checks. | Servo 1 and 2 online |  |  |
| 5 | Using the protractor, measure the position deviation from original position measured in step 2. | < 1.8° |  |  |

If the Rotator’s motors successfully perform the startup self-checks and return to the original position, the Rotator should be tested for functional performance.

A diagram of a power supply system

AI-generated content may be incorrect.

Figure 3: Mobility Test Setup

### Azimuth Mobility Test

The Azimuth Mobility Test verifies that the Rotator is capable of completely rotating 360 degrees in Azimuth. This check ensures that the Rotator will not need to “flip” when the satellite is passing directly overhead. This ensures that the signal will not be impacted while tracking directly overhead satellites.

To perform this test, the system should still be connected to power and the following steps outlined in Table 3 are to be performed.

Table 3: Azimuth Mobility Test Procedure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect Test Equipment and power supply to Subsystem Under Test (SUT) as shown in Figure 3. | N/A |  |  |
| 2 | Using a compass, take an initial measurement position measurement. | N/A |  |  |
| 3 | Using the code provided in Figure 4, send a control input that moves the azimuth to rotate 90º. | N/A |  |  |
| 4 | Utilizing the stopwatch, measure the amount of time taken to complete the rotation. | <180s |  |  |
| 5 | Using the compass, measure the azimuth angle after rotation. | 90° ± 1.8° |  |  |

A screenshot of a computer program

AI-generated content may be incorrect.

Figure 4: Azimuth Test Code

This check determines the freedom of movement of the Rotator subsystem’s azimuth direction. Upon failure, determine failure cause and document accordingly.

### Elevation Mobility Test

The Elevation Mobility Test determines the Rotator’s freedom in elevation. The elevation is where the antenna is pointed and is primarily responsible for holding the antenna at a given height above the horizon. This operation is required for proper signal strength, as deviations in elevation can lead to signal loss.

The procedure to outline this test is shown in Table 4.

Table 4: Elevation Mobility Test Procedures

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect Test Equipment and power supply to Subsystem Under Test (SUT) as shown in Figure 3. | N/A |  |  |
| 2 | Using a compass, take an initial Elevation measurement. | N/A |  |  |
| 3 | Using the code provided in Figure 5, send a control input that moves the elevation to 90º in elevation. | N/A |  |  |
| 4 | Utilizing the stopwatch, measure the amount of time taken to complete the rotation. | <180s |  |  |
| 5 | Using a compass, measure the elevation angle after the rotation is complete. | 90° ± 1.8° |  |  |

A computer screen with text and numbers

AI-generated content may be incorrect.

Figure 5: Elevation Test Code

Verification of this system ensures that the Rotator has freedom of movement in elevation, which enables for smooth movement allowing for the least amount of signal loss. Upon failure, determine failure cause and document accordingly.

### Command Response Verification Test

The command response verification test ensures that the system can receive a command and move to the given coordinates. This test verifies that the Rotator can smoothly operate both the azimuth and elevation at the same time, ensuring that there is minimal signal loss during tracking.

To perform this test, follow the procedure outlined in Table 5.

Table 5: Command Response Verification Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect Test Equipment and power supply to Subsystem Under Test (SUT) as shown in Figure 3. | N/A |  |  |
| 2 | Using a protractor, take an initial Elevation measurement at 0 degrees. | N/A |  |  |
| 3 | Using the protractor, take an initial azimuth measurement at 90 degrees. | N/A |  |  |
| 4 | Using the code provided in Figure 6, send a control input that moves 72º azimuth and 36º in elevation. | N/A |  |  |
| 5 | Utilizing the stopwatch, measure the amount of time taken to complete the movement | <180s |  |  |
| 6 | Using a compass, measure the elevation angle after the rotation is complete. | 36 ± 1.8° |  |  |
| 7 | Using a compass, measure the azimuth angle after the rotation is complete. | 72 ± 1.8° |  |  |

A computer screen shot of a program

AI-generated content may be incorrect.

Figure 6: Command Response Test Code

Completion of this test ensures that the Rotator is capable of smoothly tracking and positioning an antenna where it needs to be for obtaining a good signal. Upon failure, determine cause and document accordingly.

### Rotational Mobility Test

The rotational mobility test will determine if the Rotator is capable of operating to the limit of azimuth and elevation required for accurate satellite tracking. This test will verify that the Rotator can move to more extreme values of azimuth and elevation.

To perform this test, follow the steps outlined in Table 6.

Table 6: Rotational Mobility Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect Test Equipment and power supply to Subsystem Under Test (SUT) as shown in Figure 3. | N/A |  |  |
| 2 | Using a compass, take an initial Elevation measurement. | N/A |  |  |
| 3 | Using the compass, take an initial azimuth measurement. | N/A |  |  |
| 4 | Using code provided in Figure 8 And use Python code provided by Figure 9. send a control input that moves 172° azimuth and 86° in elevation. | N/A |  |  |
| 5 | Utilizing the stopwatch, measure the amount of time taken to complete the movement | <180s |  |  |
| 6 | Using a compass, measure the elevation angle after the rotation is complete. | 86 ± 1.8° |  |  |
| 7 | Using a compass, measure the azimuth angle after the rotation is complete. | 172 ± 1.8° |  |  |

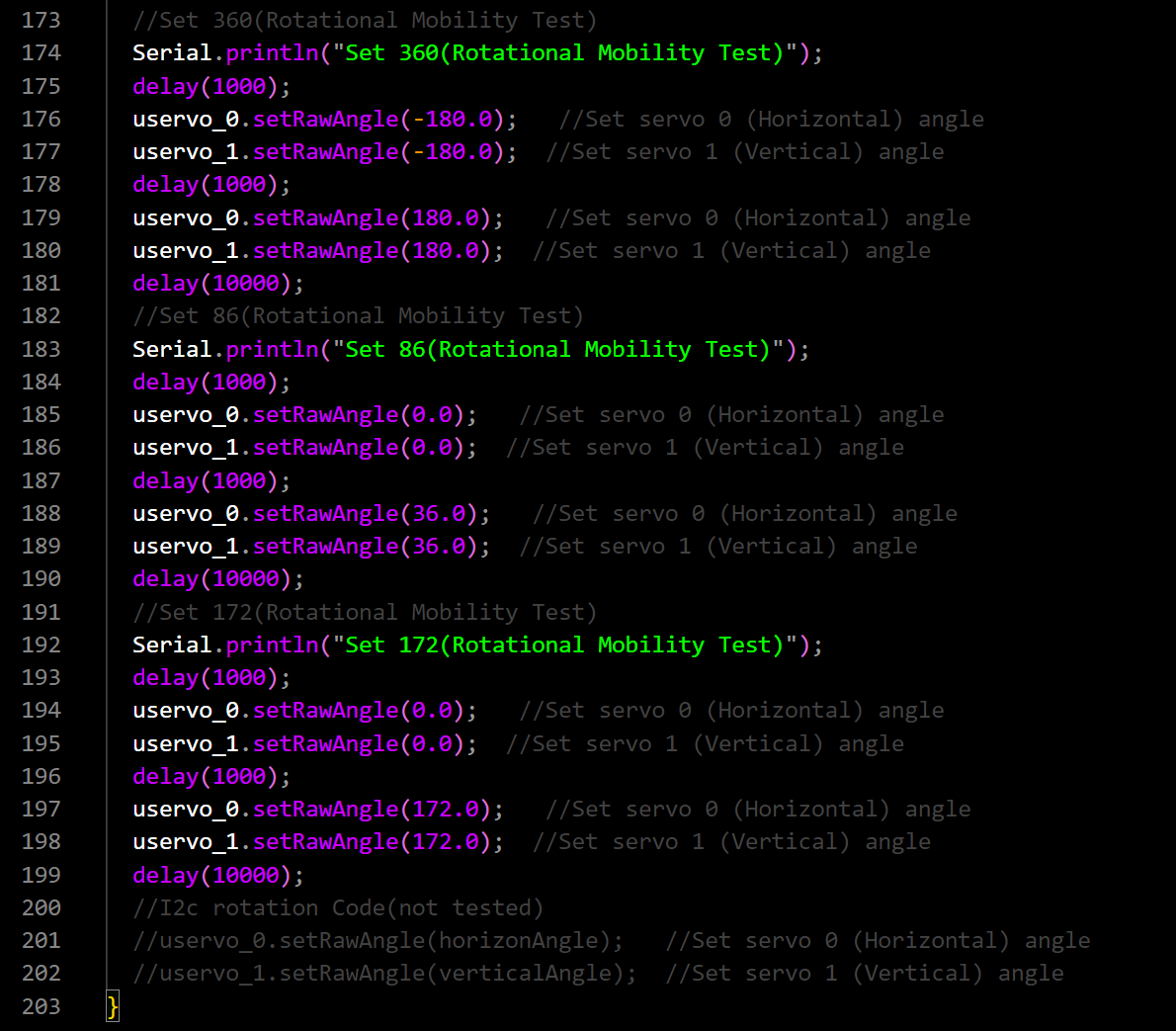


Figure 7: Rotational Mobility Test

Completion of this test ensures that the Rotator is capable of operating at the limits of required angles for tracking low earth orbit satellites. Upon failure, determine cause and document accordingly.

### Rotational Mobility Test by I2C

The rotational mobility test will determine if the Rotator is capable of operating to the limit of azimuth and elevation required for accurate satellite tracking by receiving system. This test will verify that the Rotator can receive correct messages from receive system and keep tracking right position.

To perform this test, follow the steps outlined in Table 7.

Table 7: Rotational Mobility Test

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect Test Equipment and power supply to Subsystem Under Test (SUT) as shown in Figure 3. | N/A |  |  |
| 2 | Using a compass, take an initial Elevation measurement. | N/A |  |  |
| 3 | Using the compass, take an initial azimuth measurement. | N/A |  |  |
| 4 | Using code provided in Figure 8, send a control input that moves 172° azimuth and 86° in elevation. | N/A |  |  |
| 5 | Utilizing the stopwatch, measure the amount of time taken to complete the movement | <180s |  |  |
| 6 | Using a compass, measure the elevation angle after the rotation is complete. | 86 ± 1.8° |  |  |
| 7 | Using a compass, measure the azimuth angle after the rotation is complete. | 172 ± 1.8° |  |  |

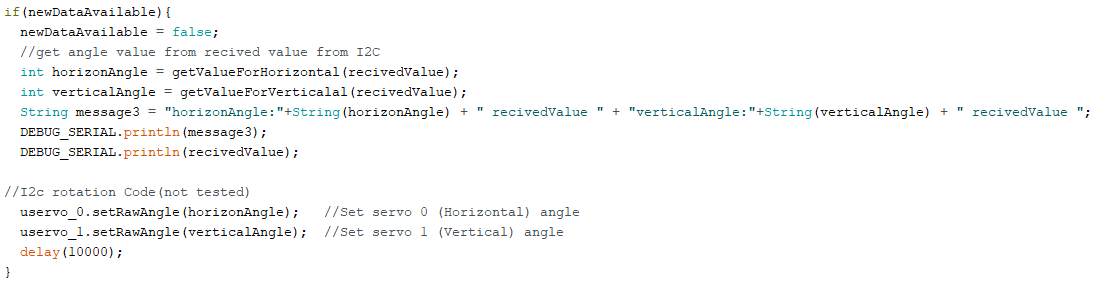


Figure 8: Rotational Mobility Test

Completion of this test ensures that the Rotator can receive satellites location from reciver for tracking low earth orbit satellites. Upon failure, determine cause and document accordingly.

### Power Consumption Test

The power consumption test verifies that the MATS has a power consumption that enables it to perform utilizing available power on board many aircraft, emergency vehicles, and maritime vessels. Table 7 outlines the procedure to successfully perform these checks, and Figure 8 shows the set equipment setup.

Table 7: Power Consumption Test Procedure

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Test/Verification Step | Expected Value | Measured Value | Pass or Fail |
| 1 | Connect Test equipment and the subsystem under test (SUT) as shown in Figure 8. | N/A |  |  |
| 2 | Connect Power Supply to SUT. | N/A |  |  |
| 3 | Using I2C, send a control input that moves 90º azimuth and 45º in elevation. | N/A |  |  |
| 4 | Record power consumed by the SUT as measured by a wattmeter | < 50W |  |  |

The power consumption test validates that the rotator operates with a power consumption that will not damage fuses, wiring and other equipment on board airplanes, emergency vehicles and maritime vessels.

A diagram of a power supply system

AI-generated content may be incorrect.

Figure 8: Power Consumption Equipment Setup

## Conclusion

The successful execution of this subsystem verification test plan ensures that the Rotator subsystem meets all functional, performance, and reliability requirements before integration into the MATS. Following the verification procedures outlined in this document, a skilled technician can systematically validate the subsystem’s compliance with the design specifications, identify potential issues, and document verification results accordingly.

## Appendix

Note:

During testing, I found that the lib provided by the servo manufacturer can no longer be used on esp32. I have contacted the manufacturer's customer service, and they still cannot solve this problem for the time being, but they will fix this bug in the future. After comprehensive consideration, since we need to use two uart ports and I2Cport to transmit and receive data, I chose the arduino mega r3 development board. The only difference from the original solution is that mega r3 requires 7-12V power supply. So in the future, if the servo manufacturer can fix the lib, we may still use esp32.

Shown below is the test code program.