Performance Analysis Report

ARES Performance Analysis Report

4/14/2024

Performance Analysis Report

Prepared by: ____________________________________

George Garcia
Chief Engineer, ARES

Reviewed by: ____________________________________

Manan Khandelwal
Project Manager, ARES

Approved by: ____________________________________

Philip Mathews
Program Manager, AGSL

Performance Analysis Report

REV	Date	Description of Change	Prepared	Reviewed	Approved
0	6/4/2024	BASELINE RELEASE	GG	MK	

Performance Analysis Report

Table of Contents

Performance Analysis Report

1 Executive Summary

The ARES (Autonomous Roving Exploration System) rover was one of two teams repeating AggieSat's 2022 - 2024 SCAMPS mission challenge. The mission remained the same: navigate toward a radio beacon transmitting at a specific frequency while avoiding obstacles, all on an undisclosed terrain meant to simulate extraterrestrial conditions. While the rover passed all subsystem tests and almost all integrated tests, lapses in final-stage full-system testing resulted in an incomplete mission. While the rover achieved motion in a straight line and collected data, it failed to demonstrate key elements of the mission, such as detecting the beacon and actively avoiding obstacles. The rover was also unable to collect a full five minutes of data. This report explains ARES' successes and failures, and details how future attempts at similar missions can be made more successful.

2 Introduction

2.1 *Background*

After an unsatisfactory performance at the 2024 SCAMPS showdown, the ARES team was one of two teams formed to attempt to repeat the intended mission to demonstrate proper systems engineering principles and build a functioning project.

As mentioned earlier, the mission remained unchanged from the previous year. The mission of ARES was to detect the heading of a beacon transmitting at 433 MHz and avoid obstacles intentionally placed in the beacon's path. After the conclusion of the traversal stage, the rover was designed to collect local temperature and wind speed data for five minutes before transmitting the collected telemetry back to the ground station. Additionally, success depended on physical design constraints, including mass, dimensions, and cost.

While attempting the mission, ARES failed to detect the beacon's heading and instead traveled in its original direction upon activation. It also appeared to veer around an obstacle, but this was likely due to the slope of the ground that steered the rover away from the obstacle. Towards the end of the mission, the rover moved towards an adjacent concrete pavement and got stuck on the pavement's ledge, which activated its backup measures after three minutes. Due to the lack of motion, the rover prematurely changed states to the sensor collection state, where it attempted to collect environmental data but failed to recover all 300 packets due to faulty wiring and data corruption. The data was also stored locally on the microcontroller instead of transmitting it to a ground station, since the radios failed to connect on the day of the attempt.

After comparing the mission requirements with the rover's performance, the ARES team can confirm that the rover **failed** to accomplish its mission. This document explains the team's successes and failures throughout the mission timeline and describes any additional steps that could have been taken to improve the rover's performance.

2.2 *Timeline and Budget*

The development period for the ARES system was approximately 7 months, as specified per the SCAMPS 2023 overview. The ARES rover had a total cost of $776.93, per the $1000 limit levied by the SCAMPS mission constraints.

3 Systems Engineering Breakdown

3.1 *ARES Team Organization*

The team was organized into five subteams: TMS (Thermal, Mechanisms, and Structures), EPS (Electrical Power Subsystem), CDH (Command and Data Handling), GNC (Guidance, Navigation, and Control), and COM (Communications). The members on each subteam are listed below.

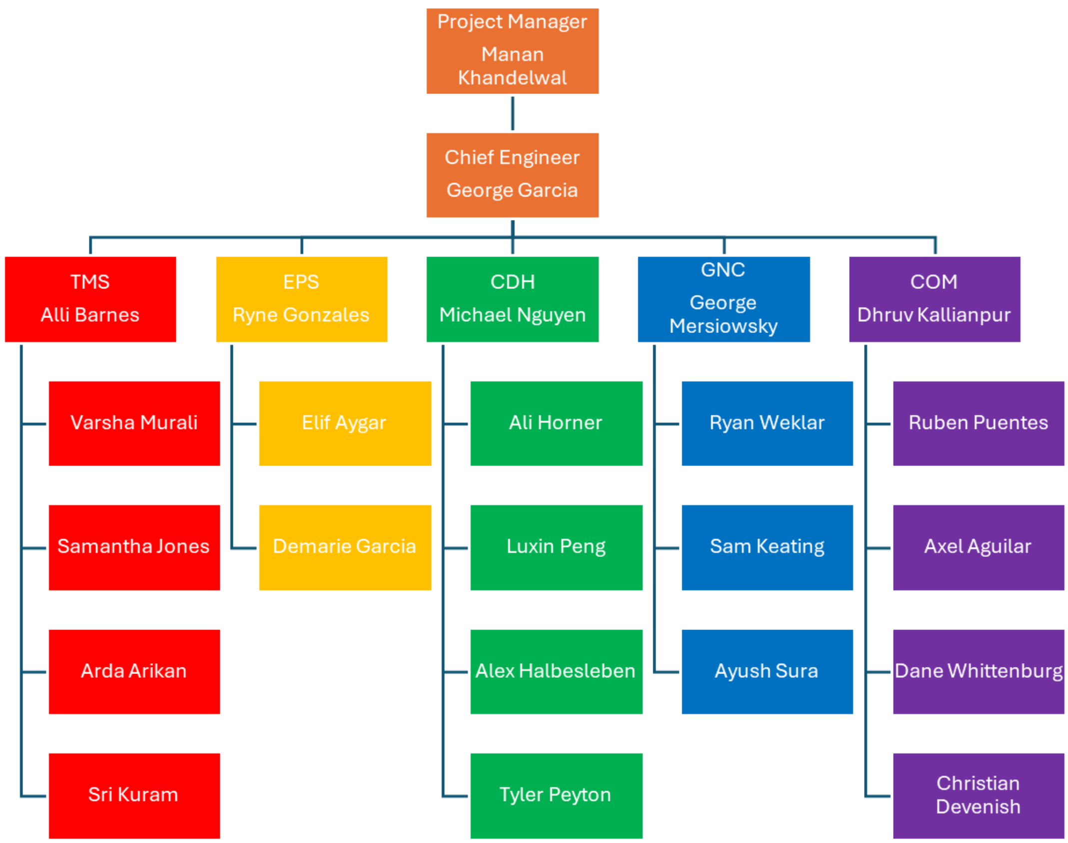

Figure 1: ARES Organizational Chart

The four tiers of the aforementioned organization chart are described below:

1) Project Manager: The PM's role was to lead the project's direction, track the budget, handle team management affairs, and enforce the foundations of systems engineering during the project development cycle.
2) Chief Engineer: The CE's role was to lead the technical aspects of the project and ensure steady communication between subteams about changing subsystem

designs. The CE was also primarily responsible for reviewing documentation and suggesting test plans and procedures for each subsystem.

3) Subteam Leads: Each of the five subteam leads was responsible for leading the design of their respective subteams and delegating responsibilities among 2-4 members and shadows. They were the primary point of contact for enforcing inter-subteam communications and ensured documentation was delivered to the team management on time.
4) Subteams:
    a) Thermal, Mechanisms, and Structures (TMS): Responsible for developing the chassis of the structure, communicating the placement of components on the rover, and ensuring the physical constraints were met.
    b) Electrical Power Subsystem (EPS): Responsible for selecting electronics and providing sufficient power to all electrical components on the rover, including but not limited to motors, sensors, and communication equipment.
    c) Command and Data Handling (CDH): Responsible for writing the primary software driving the rover and facilitating the seamless completion of the state machine. They were also responsible for developing a ground station to process incoming telemetry and environmental data from the rover.
    d) Guidance, Navigation, and Control (GNC): Responsible for selecting and programming the components required to implement obstacle avoidance capabilities.
    e) Communications (COM): Responsible for selecting and programming the necessary components for the beacon detection and ground station data transmission stages.

3.2 *V-model Description*

Like the previous projects, the chosen systems engineering framework was the V-Model.

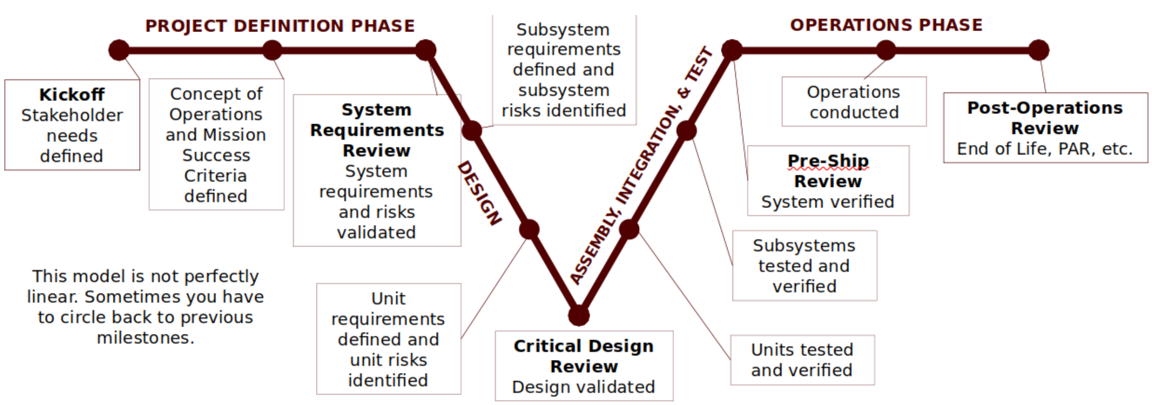

Figure 2: ARES V-Model for system design/development

The project definition phase was primarily focused on developing and refining this year's Requirements Verification Matrix, or RVM. The RVM remained fairly unchanged from last year's requirements due to an unchanged mission, but additional requirements were levied to ensure the team's success and guide the rover's early design.

The design phase was defined by developing each subsystem's trade studies, which cross-referenced the RVM. These trade studies helped each subsystem determine the best solution to each requirement via direct comparison, and each trade study was reviewed by the team management to ensure the requirements of other subsystems were considered. The solutions from most of the trade studies were preserved throughout the design phase, although issues with components like the NRF24L01 radios prompted changes across multiple subsystems to accommodate replacement components. The newer components were selected without a trade study, but the decision was based on prior experience with those parts, online reviews, reliability, availability of documentation, and margins in the cost/mass/power budget. There were also several setbacks for the COM subteam due to the lack of a suitable antenna within the physical constraints of the rover. This was eventually overcome when the team intentionally surpassed the physical 40 cm dimensions to accommodate a directional antenna that simplified our beacon-finding procedure. Despite some issues during the design stage across all subteams, constant communication and the upkeep of documentation helped keep the team mostly on schedule.

The bulk of the scheduling problems took place during the integration and testing phase. The team's most prominent issue was the mishandling of parts like the Rev C Wind Sensor, DC motors, etc, which required us to consistently order new parts. Additionally, the aforementioned design changes for one subteam also prompted design changes for another subteam, so the team had to revert to older stages several times before moving forward. Despite the timeline issues, each subsystem ran at least one integrated test that proved the functionality of their system, and some subteams were also able to run integrated tests with others. This was the closest the team got to delivering a fully functional system, although this level of integrated testing should have happened sooner in the timeline. During the last week, last-minute part failures and delayed part delivery pushed full system testing to the last night, where the rover had most of its systems functional. The key point of failure was the final mounting of electronics, which was pushed to hours before the deployment because the team was testing all of the electronic systems. This handover was delayed because the team did not want to risk losing electronic components after mounting everything, which would have almost guaranteed a system failure. During the final mounting, the power connection between the battery pack and the PCB was damaged slightly, which is discussed in detail below. Solutions to fix this timeline challenge are also discussed below.

4 The Envisioned vs Actual System

4.1 Brief Description of Concept of Operations

The ARES system concept of operations consisted of five operational modes illustrated below:

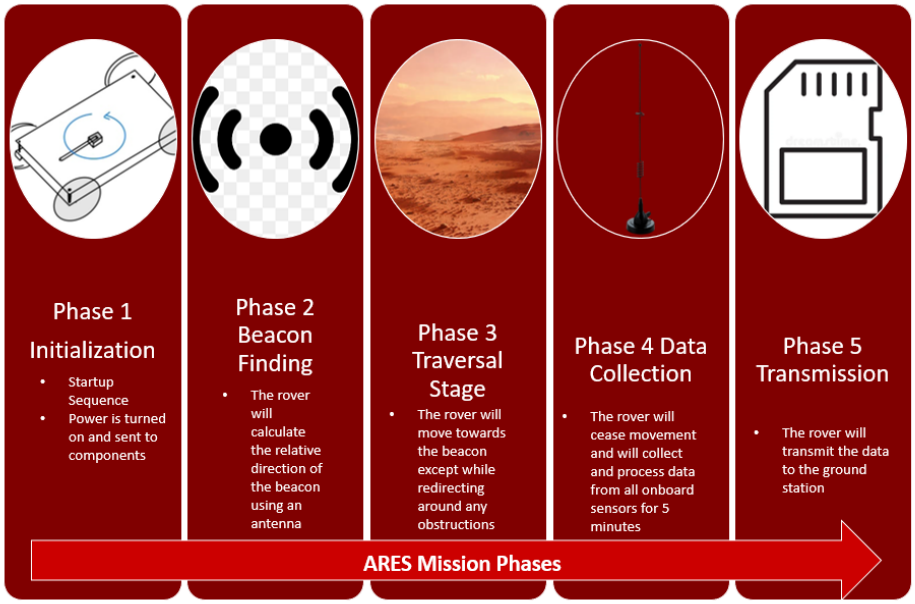

Figure 3: ARES Concept of Operations

The ARES CONOPS provides a detailed description of the ARES operational modes, as well as the specific subsystem and system-level design.

4.2 *Design Considerations from SCAMPS 2024 Performance*

Each team made several design considerations to improve on the issues from the 2024 showdown.

4.2.1 Thermal, Mechanisms, and Structures (TMS)

Last year, both rovers were largely unable to move due to being placed in unexpected terrain and not having enough power in the motors to traverse the field. TMS helped fix this issue by designing wheels with a larger diameter and deeper grooves to increase the ground clearance for GNC and improve traction, respectively. Additionally, the configuration of the rover was changed from one to two layers to attempt a better electronics wiring configuration with separate control and sensor sections, but this was not executed as envisioned in the end.

4.2.2 Electrical Power Subsystem (EPS)

Concerning the EPS subsystem, many design changes were made after learning from SCAMPS 2024. A PCB was used to make the wiring more organized to easily assemble and debug any electrical issues, and wires were color-coded for identification and handling. To solve the aforementioned power issue, a stronger set of motors was used to provide the drivetrain with sufficient power. Another prominent issue was electrical failure on the integrated circuitry for the GeoRGE rover last year, so safety precautions were also taken during the development of the rover. For example, fuses were used to incorporate safety features to protect components from shorts and overcurrent, and all members were required to use ESD mats when handling sensitive electrical components.

4.2.3 Command and Data Handling (CDH)

The biggest issue with the CDH team from last year was the lack of testing done on the rover. To fix this, each CDH member specialized in different functions, such as tackling the ground station, motor software, state machine, Git workflow, etc. The testing problem was solved by introducing configurations, or ‘configs’ within the VSCode/PlatformIO/GitHub system, which allowed independent codebases to be developed by different members without the mess of uninformed Git branches. This allowed for a faster testing process that was further enhanced with version control, allowing CDH to complete unit and subsystem tests much faster than last year’s schedule.

4.2.4 Guidance, Navigation, and Control (GNC)

To improve GNC’s functionality, the motivation behind the subsystem’s design was to keep the system simple. To facilitate this, GNC used two ultrasonic sensors with readily available test software to develop the obstacle avoidance state, and conducted tests to determine the optimal placement of these sensors.

4.2.5 Communications (COM)

Based on previous experiences, COM avoided interferometry as a method of direction finding, learning that it would require too much precision to work accurately, so a directional antenna was used instead. Furthermore, COM replaced the planned SDR in favor of transceiver units, learning that the low clock speed of an Arduino leads to difficulties in integrating an SDR.

4.3 Comparison between the envisioned subsystem design and the subsystem at deployment

4.3.1 Thermal, Mechanisms, and Structures (TMS)

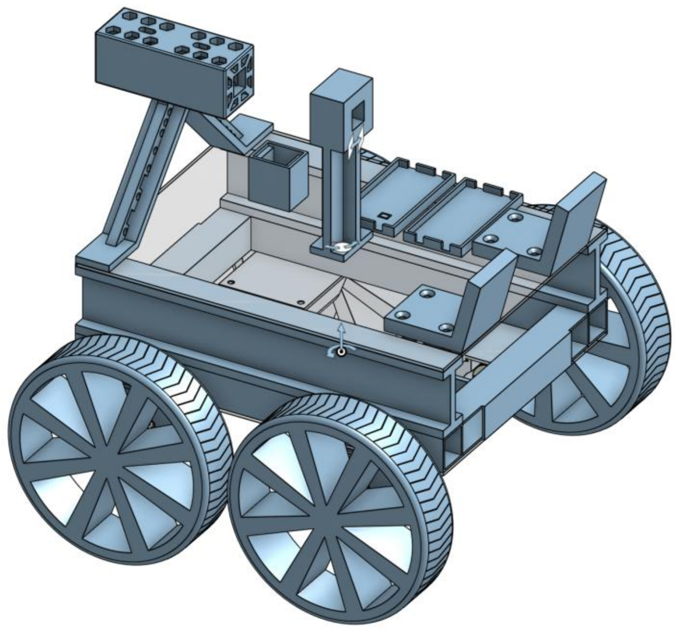

Figure 4: Final CAD model of rover

The end design was very close to the initial plan, except for the wheels. The wheels had initially been planned to be RC car wheels, but based on feedback from GNC, they were adjusted to be 3D-printed wheels to increase the height of the rover. There were some initial issues with mounting the 3D-printed wheels. The antenna mount was also completely redesigned after COM replaced the omnidirectional antenna with a directional antenna.

4.3.2 Electrical Power Subsystem (EPS)

In the fall semester, EPS created the wiring diagram shown in Figure 5 based on the parts each subteam had originally selected for this challenge.

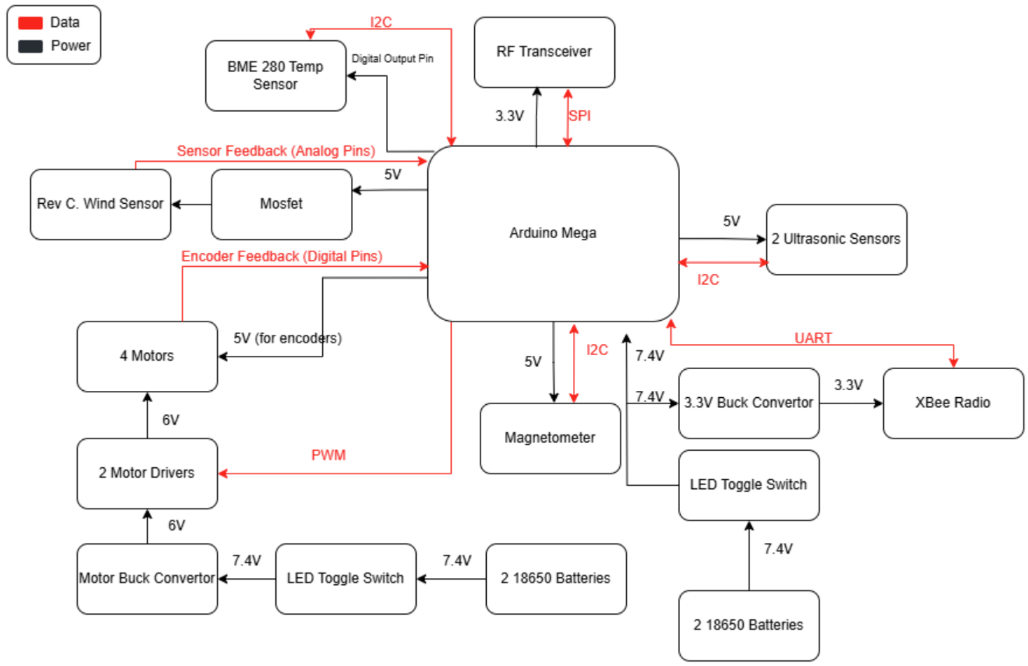

Figure 5: Planned Wiring Diagram (Fall Semester)

After each subteam finalized their respective component selections, the wiring diagram shown in Figure 6 was developed to serve as a comprehensive reference for the system's electrical layout. While this diagram does not fully represent the final implementation on the ARES rover, it provided the foundational framework upon which the intended wiring configuration was based.

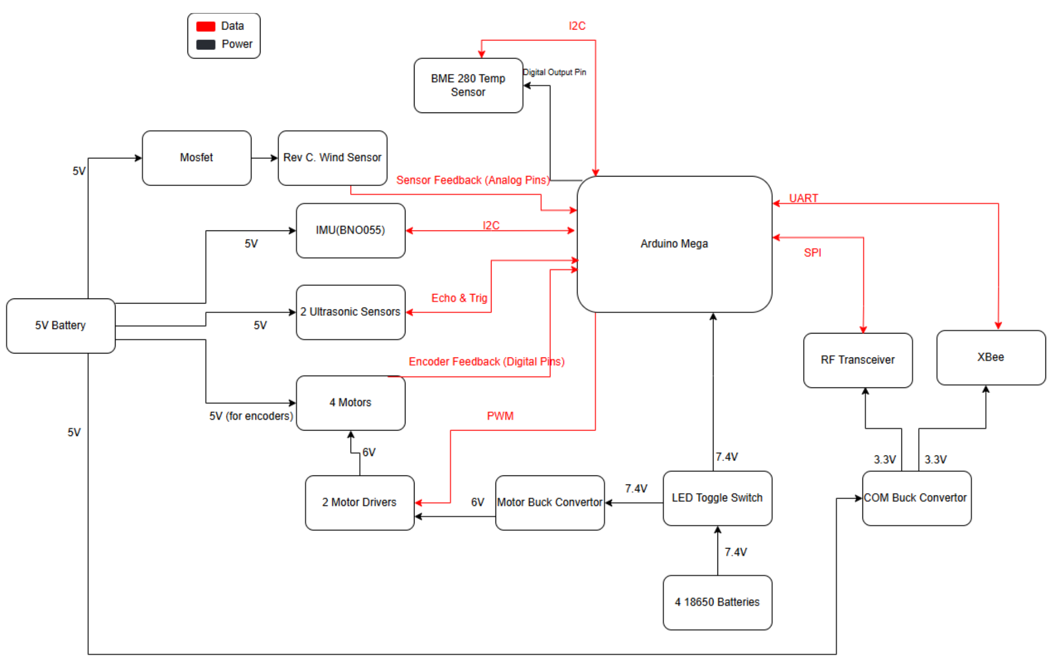

Figure 6: Intended Wiring Diagram

The original planned design (Figure 5) employed two separate 2-cell 18650 battery packs, each delivering 7.4V and powering distinct parts of the system through individual LED toggle switches. This setup added redundancy and allowed selective control of subsystems, but it also increased wiring complexity and introduced challenges in power distribution and balancing. In contrast, the final planned design (Figure Y) transitioned to a simplified and centralized power scheme by using a single set of four 18650 batteries in series, providing a total of voltage. This single source was then stepped down using dedicated buck converters to supply the needed voltage to various components, including motor drivers, sensors, and communication modules. Additionally, a 5V battery was introduced to power lower-voltage components separately, isolating sensitive sensors and encoders from high-power fluctuations.

Based on the wiring diagram above, the planned design utilized a custom PCB to streamline the rover's wiring. This approach was intended to ensure that each component received the necessary voltage and current requirements, while also enabling organized placement and reliable connections to the designated Arduino pins. Figure 7 below shows the schematic intended for the PCB.

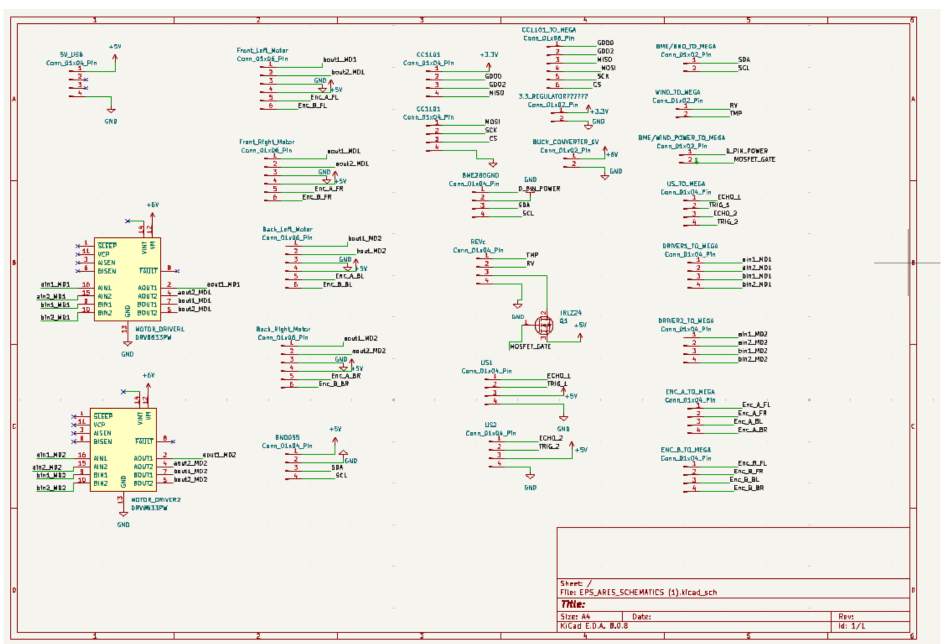

Figure 7: PCB Schematic

Last-minute modifications were required on the morning of the deployment due to connection issues with the 5V battery supplying power to the sensors. As a workaround, the sensors on the top level of the rover were disconnected from the PCB and rerouted using jumper wires and breadboards to interface directly with the Arduino. The components that were wired to the Arduino included the Rev C wind sensor, both ultrasonic sensors, the BNO055, the XBee, and the CC1101. Other components, such as those involved with the drivetrain, remained connected to the PCB as originally designed. Despite these changes, the final system deployment remained mostly consistent with the overall EPS architecture.

4.3.3 Command and Data Handling (CDH)

The envisioned state machine, which dictates the rover's behavior, shown in *Figure 5,* was much simpler than its final form. The original state machine featured 5 states:

1. Beacon detection state (S0) - perform procedure to determine the beacon heading.
2. Traversal state (S1) - move directly along heading.
3. Obstacle avoidance state (S2) - perform procedure to maneuver around obstacles when detected.
4. Data collection/transmission state (S3) - collect and send the required environment data.
5. Mission complete state (S4).

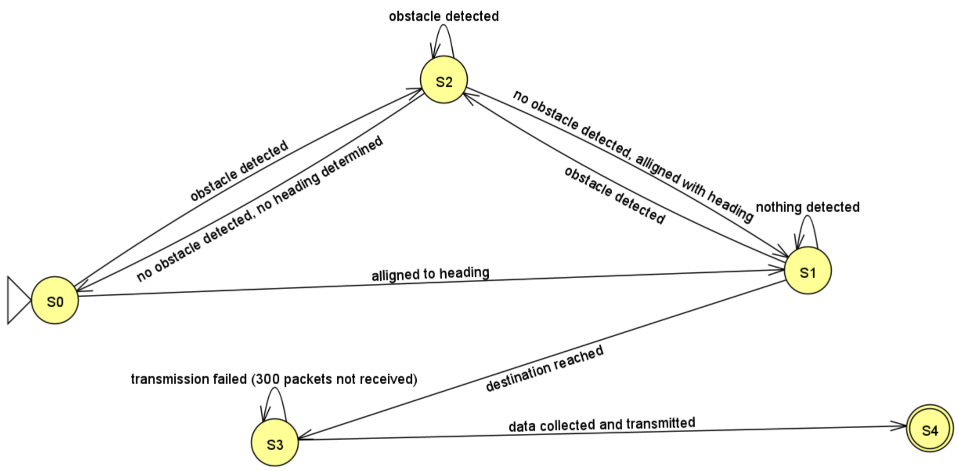

Figure 8: Envisioned state machine diagram.

The CDH team also planned for unexpected conditions and failsafes. For example, the team initially planned to have a traversal substate to handle situations where the rover was stuck in terrain. Although this code was developed, it could not be tested and implemented due to time constraints.

Another failsafe retroactively implemented was a condition to start data collection if less than ten minutes were remaining in the mission duration. This allowed the team to achieve minimal success by collecting and transmitting the required 300 datapoints if the rover was unable to get to the beacon location. These changes added to the complexity of the state machine, leading to its final form shown in *Figure 9*.

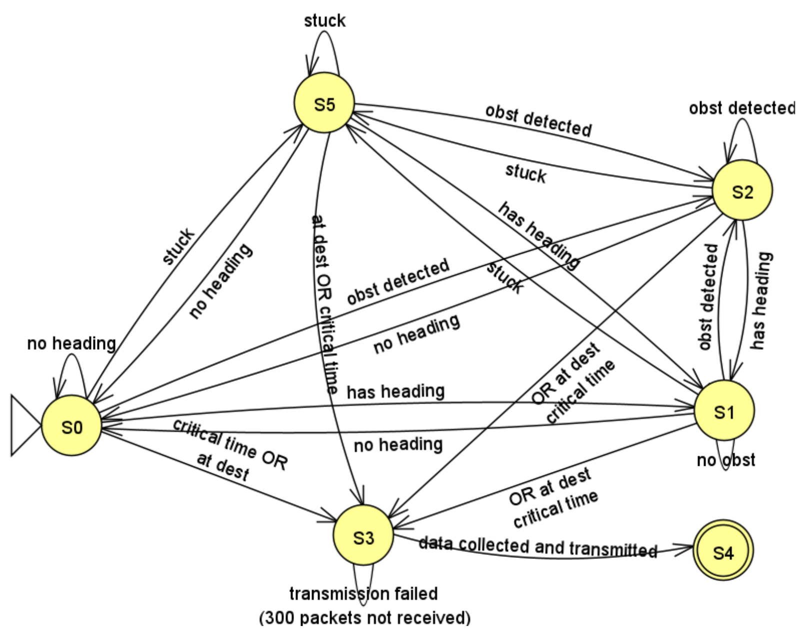

Figure 9: Final state machine diagram.

During the deployment, however, time constraints and uncertainty about the functionality of beacon finding and obstacle avoidance led to the state machine being heavily augmented for operations. Issues with mission-critical components discussed in later sections rendered the beacon finding (S0) and the obstacle avoidance (S2) states obsolete. Only the beacon-finding state was manually disabled; the obstacle-avoidance state was never invoked since the rover did not encounter any naturally occurring obstacles during operations. Due to unresolved issues and the delayed implementation of the stuck (S5) state, it was disabled before operations as well. As a result, the code running on the rover during operations was a heavily stripped and simplified version of what had been written and tested up to that point.

4.3.4 **Guidance, Navigation, and Control (GNC)**

The subsystem at deployment was an incomplete version of the envisioned subsystem, but not a changed subsystem. Due to the compressed system-level testing timeline, much of the code developed by GNC was not in use. Rather, a simplified version of the code was deployed, as it was all that had been satisfactorily tested by the time of the deployment. Specifically, the code for beacon finding was essentially unreachable, and the obstacle avoidance code was never seen because there were so few obstacles on the field at the time of the deployment. The magnetometer was still in use for beacon detection and orientation maintenance. Between the software that was tested and the final software at the time of deployment, the only change made was reducing the detection range of the ultrasonic sensors. This was because the directional antenna was discovered to interfere with the vision cone of the ultrasonic sensors, which returned false positives. As a result, this revision was never tested, and the impact on performance was never measured since no naturally occurring obstacles were in the rover's path.

4.3.5 **Communications (COM)**

While the subsystem was working as envisioned the night before the deployment, there were difficulties in integration as the final mounts and components were being placed on the rover. After this final integration, the rover did not receive a signal from the beacon, and the XBee no longer had a consistent connection to the ground station. These failures are described in detail in Section 5. Although the ideal positioning and configuration of the antenna remained the same as intended, the full functionality of the COM subsystem was eventually compromised to achieve minimum success moments before the rover's deployment.

5 Causation for Failures

5.1 *Thermal, Mechanisms, and Structures (TMS)*

TMS's main issue was the design timeline, since design changes with other subteams heavily impacted the layout of components on the two platforms. Full system testing could not be completed because the components were not fully mounted until the night before the deployment, but this could not have been avoided due to the rate of component failures that other subteams faced. The team also initially overlooked mounts for each sensor, but this was solved quickly.

Another consistent issue TMS faced was the design of the wheels. After the performances in SCAMPS 2024, the original design idea was to use off-the-shelf rubber wheels, intended for use with hobby rovers and R/C cars. Unfortunately, the wheels ordered were too small to maintain the right ground clearance for GNC's ideal performance, and the wheel hubs were also not the right shape for the motor shafts. The initial solution was to 3D-print different iterations of wheels, separated by radius to fit GNC's specifications. Once a wheel was decided and all four wheels were printed, the initial build configuration was completed for the drivetrain, but the team faced issues with wheel wobble. Different solutions were explored, such as using a different set screw, changing the shape of the wheel hub, etc, but the team finally solved the problem by ordering metal wheel hubs and embedding them within the wheel. This still caused issues with slippage and the wheels falling off, but the fix for these problems was easier than the wobble. While this problem did not directly impact performance during the deployment, it took away significant time from the project timeline that could have been used toward the TMS/EPS integration phase.

5.2 *Electrical Power Subsystem (EPS)*

The EPS subteam primarily struggled with the PCB. We had a weak connection from the USB breakout board to the PCB, which caused inconsistent power delivery to the PCB. This resulted in the team rewiring the rover away from the PCB directly to the Arduino Mega before the operation, which likely did not provide sufficient power to all of the components, especially the CC1101. The transceiver either did not receive enough power, or was fried since the component was rated for 3.3V but was connected to the 5V outlet. The justification for this was that the Arduino Mega simply did not detect the transceiver when plugged into the 3.3V outlet, but did so when connected to the 5V outlet.

Furthermore, EPS also had some issues with time management. The team had several components to test, and components like the wind sensor, the motor driver, and motors malfunctioned, warranting replacements. The system was tested thoroughly but not efficiently, leading to delays in other subteams' developments.

The last-minute decision to design and order a PCB was also a cause of failure, since it also deviated the team from the standard component testing timeline. The reason for this decision was to organize the wiring for each component and reduce the odds of failure, but this adversely affected the system and led to a bottleneck. Ultimately, the PCB power failure proves that this decision was not thoroughly planned, and the system could have achieved key requirements without it.

5.3 *Command and Data Handling (CDH)*

Every unit of the final CDH subsystem was successfully tested on a reasonable timeline. Several integration tests were also performed successfully, including obstacle avoidance, beacon detection, course correction, wireless data transmission to the ground station, and the ground station itself. However, CDH struggled to perform system-level tests because the full system was not test-ready until the days leading up to the deployment. As a result, errors that could have been fixed, such as bugs with beacon finding, were left unresolved, and large swaths of code were removed from the codebase. On a more appropriate timeline, CDH would have been able to debug these errors and use more of the code that the team had written up until that point.

5.4 *Guidance, Navigation, and Control (GNC)*

While the subsystem itself was tested independently, GNC never had the opportunity to test at the system level due to timeline issues. At the unit level, the ultrasonic sensors were tested and proved reliable at detecting large obstacles at short to medium distances, which matched expectations. The magnetometer was also thoroughly tested and proved extremely reliable at outputting its current orientation. However, the system was not fully functional until shortly before deployment, and subsystem-level tests only validated the design of the control algorithm, but not its implementation on the rover. To solve this problem, CDH adapted a different version of GNC's code at the last moment which resembled the control algorithm, but displayed delayed reactions to intentionally placed obstacles.

During deployment, the placement of the ultrasonic sensors was an unforeseen issue. GNC's original design was to angle the ultrasonic sensors down at a 15-degree angle, but a separate mount angled at 5 degrees upwards was printed and installed. This design change was made because of the lack of system-level and environmental testing, which meant the detection cone of the sensors was not tuned. Unfortunately, the new mount unexpectedly threw false positives by detecting the tip of the directional antenna. This problem was mitigated by slightly altering the mounted angle of the sensors, but this correlates to a slight decrease in the observable range of the sensors.

Additionally, as was mentioned in section 4.3.4, much of the GNC code had been commented out during last-minute system testing since the rover could not accurately rotate, and was not restored before the deployment.

5.5 *Communications (COM)*

Testing revealed that while the CC1101 transceivers could detect a signal from the beacon, the signal strength was not as consistent as envisioned when integrated with the rest of the system. Downlinking at the maximum range was successful, but there was no testing on downlinking real experimental data at that maximum range, which led to issues during the final testing. Once integrated, the antenna struggled to pick up the beacon's signal, contradicting all the promising testing seen beforehand. This was a result of the CC1101 either not receiving enough power or being fried, which is detailed in Section 5.2.

Additionally, the XBees were tested thoroughly with data transmission on the morning of deployment, but there was no signal after the final component mounting. The team attributes this to similar reasons as the CC1101. As a result, neither of the COM systems worked during the rover's deployment.

6 Systems Engineering Efforts/Lapses

The timeline for this project was designed with buffers in mind, and most of the preliminary design stage was on track during the fall semester. Additionally, all subteam leads and the team management made active efforts to encourage inter-subteam communication via the following:

1. Start of meetings: The day's agenda was shared, and any preliminary questions were answered.
2. Middle of meetings: The team management circled between different subteams to discuss progress, provide guidance, and share other subsystems' progress to ensure all design changes were tracked.
3. End of meetings: Each subteam shared their progress from the day, and any questions were answered.
4. Outside of meetings: Meeting results/agendas were posted, and the team management reached out to subteams individually for mid-week deliverables. A team calendar was also created around March 2025 to encourage team members to attend subteam meetings. Unfortunately, these efforts were sporadic, and communication was hindered by reasonable commitments to affairs outside of the team.

ARES also trained several shadows across both semesters, which helped bring fresh design perspectives to the team and accelerated the team's development as well as individual skillsets. The team also took the opportunity to introduce shadows to the lab's core systems engineering principles, which shadows were able to practice by looking through their subteam's documents and updating them on an as-needed basis. Unfortunately, the distribution of work was oversaturated at some points in the system's development, which led to dead zones, a lack of productivity, and increased difficulties with communication. Nonetheless, each shadow played a vital role in the design and testing of the rover.

The team also did a commendable job updating its documentation by the end of the project. To help the approval cycle, the team used a Document Review Queue managed by the Chief Engineer and Project Manager, which was routinely checked for updates. New documents and revisions were controlled and approved here, and any edit suggestions were checked and reworked by subteam members. A key lapse in documentation occurred during trade studies, where the purpose of a trade study from a systems engineering standpoint was forgone in favor of past experiences, leading to biased decision-making. In the future, this can simply be corrected by discouraging casual selections and testing the results of trade studies through an informal CoDR (Concept Design Review).

The flow of system testing from the component to the full system was an undeniable lapse in systems engineering for several reasons. First, members were trained on electrical safety procedures, but the use of an ESD mat was not thoroughly enforced by the team management. This can be corrected by increasing the availability of ESD mats in the lab, since that was a factor in not using them. There was also no accountability in following the test procedures as they were written. In the interest of time, many test procedures were overlooked in favor of quick scripts found online. While this worked and helped accelerate the timeline, it did not align with systems engineering principles. One way to correct this is to write an abstract test procedure specifically for components with extensive online documentation, such as the BNO055, BME280, etc. This does not apply to components that must be unit tested in conjunction with other components, such as the CC1101, which requires a connection to the beacon or another transceiver.

Additionally, the team never revisited the RVM after selecting components to ensure that the theoretical system design met the requirements originally set by the team. While some of these could only be proven through testing, the formal procedure was never completed, and the team was left uninformed about alternate solutions to the system design. This lack of information during testing directly caused late design choices. For example, the decision to order a PCB should have been rejected since the tradeoff between time and organization was not worth it.

The late design choices directly interfered with the bottom-up flow of testing in the spring semester, but the separate timelines for each subteam also exacerbated the problem. For example, COM had not finalized a beacon detection system until February, which is when other teams were well into component testing. The team tried to cope with separate timelines for as long as possible with built-in timeline buffers, but the bottlenecks created due to this directly contributed to the system's failure at the end.

7 Human/Leadership Factors

7.1 *Team Management*

As mentioned earlier, the team management emphasized open lines of communication above all concerns during meetings. This allowed most members to be informed of design changes, which helped other subteams respond accordingly and build the system from the ground up. However, there are a few areas of improvement:

1. Enforcement of proper systems engineering practices, like design reviews, trade studies, and requirement validation. This was discussed in Section 6.
2. Rapidly authorizing design changes without risk and decision rationale - more thought into why changes are warranted and their potential impact could have informed decisions better, especially in the spring semester.
3. Organization of information - The team management could have done a better job routinely organizing leadership meetings, having shadows read documentation before assisting with design, and minimizing redundancy within documents.

7.2 *Subteam leads*

ARES' subsystem leads greatly supported the team management in accomplishing many of the leadership goals throughout the year. Leads promptly responded to concerns, guided shadows with experience, and regularly hosted subteam meetings that resulted in increased team productivity during the week. The following are subteam-specific areas of improvement from the leads' perspectives:

1. TMS: Could have pursued a slightly more aggressive schedule during the early design phases, and should have been a little more thorough with the CAD design process to avoid last-minute tweaks. In the future, TMS should wait until after the theoretical system design to determine a development schedule for CAD drawings and manufacturing.
2. EPS: Should have decided to order a PCB in the fall, instead of reviving and implementing the idea in the spring. Also could have created a tool to communicate the testing schedule to other subteams more effectively.
3. CDH (PM's perspective): Did a good job allocating tasks among members, could have improved on communicating progress to EPS and GNC leads to ensure the system was fully integrated (this is true for all of ARES' leadership, though)
4. GNC: Should have pursued a more aggressive schedule, which led to lapses in productivity. PM's perspective - even though the end system was not integrated properly due to the overall rover testing schedule, this was not GNC's fault. They could even have considered a more challenging approach to the GNC algorithm, which could have led to more skill development for GNC members. Did not feel like GNC members took away much learning from this project.

5. COM (PM’s perspective): Did a good job seeking external sources of information for the COM system design, and made the right call to move to the directional antenna. However, this choice should have been completed in the fall semester, and the timeline was pushed back much further than anticipated. Could have accomplished more testing as a result, but still made it very far given the short time they had left in the spring.

8 Lessons Learned

While the ARES system ultimately did not receive full mission success, the team identified several areas for improvement. These lessons learned are valuable both to the ARES team and future teams trying to achieve similar goals.

2. **Design Changes/Trade Studies:**
   There were too many design changes made in the spring semester that should've been considered and finalized in the fall, such as COM’s antenna and EPS's decision on whether to use a PCB. Because of these decisions, it created bottlenecks in system integrations, resulting in the team having to pull all-nighters to finish the integration. This could've been solved with better trade studies or sticking with the original design. As well as include more margin of error within the design.

3. **Inter-subsystem Communication:**
   While it was a much better improvement compared to the previous projects, there were times when some subteams were under the wrong impression of what a subteam was working on because they didn't communicate it. This could be implemented through separate team lead meetings or allotted times for inter-team meetings during workdays.

9 Conclusion

The ARES project faced numerous challenges throughout its development, most of which came from timeline issues because of part failures and last-minute design changes, and integration. However, through the reflection and analysis of the system's shortcomings, the team believes that the core goal of SCAMP was accomplished, which was to teach students the fundamentals of systems engineering principles through hands-on design, testing, and integration, which was ultimately achieved. Despite the project's failure, lessons learned have provided valuable experience that each ARES member will carry into future endeavors.

10 References

[1] SCAMPS 2023 Overview
[2] ARES-CONOPS
[3] ARES-RVM