Performance Analysis Report

GeoRGE

AggieSat Laboratory

4-28-2024

Table of Contents

1 Introduction

1.1 Background

The Geographical Radio Guided Exploration (GeoRGE) Rover project team competed in the Student Competition for Autonomous Measurement of Planetary Surfaces (SCAMPS) over 16 months. SCAMPS was created to give students hands-on experience with systems engineering on a design, build, and test engineering lifecycle. The competition consisted of two teams that were graded on their full systems engineering process.

The final showdown took place on April 7th, 2024, in the Brazos Valley. The mission defined in the SCAMPS Overview was to autonomously navigate to a beacon with a known signal while avoiding obstacles. Once reaching the beacon, the vehicle was to then record environmental conditions for 5 minutes and transmit the data to the ground station. [1]

During the showdown, the GeoRGE rover failed to complete its CONOPS and mission success criteria. The rover was unable to navigate to the beacon or send data to the ground station.

This document will provide an in-depth analysis on the problems that were encountered, as well as the causes behind them.

1.2 Timeline and budget

Though intended to be a two-semester project, the competition was moved due to the teams not being far enough to compete. The organizers of the competition pushed the competition back, which led to the duration of the project lasting 16 months. The system, excluding the computer used as ground station, cost $866.50 to produce. This was $133.50 less than the given SCAMPS constraint of $1000.

2 Team Systems Engineering

2.1 Organization

The team organization for the project can be seen in ***Figure 1***. Due to graduations and members leaving, many subteam lead appointments changed through the duration of the project.

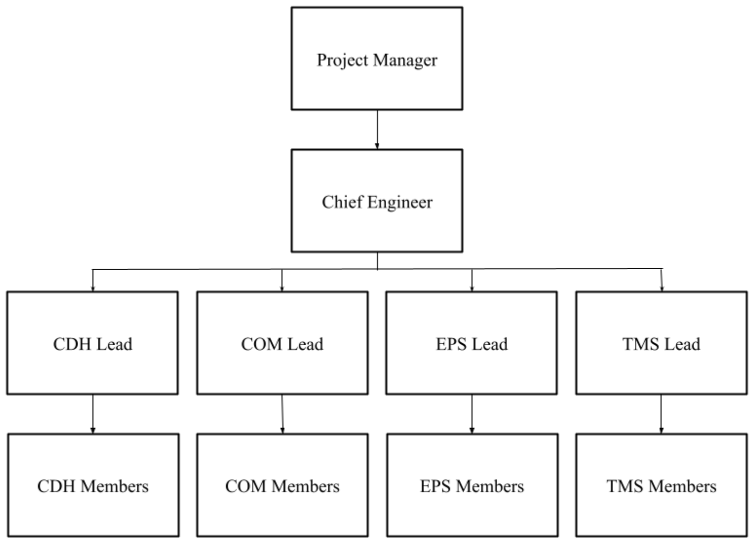

Figure 1. GeoRGE Organizational Model

The team consisted of four teams, with a team lead in charge of each: CDH, COM, EPS, and TMS. CDH was responsible for the command subsystem, which consisted of data collection and electrical and navigational control components. COM was responsible for the communication system, which consisted of both on-rover and ground station communication subsystems. EPS was responsible for the electrical subsystem, which consisted of power distribution and budgeting. TMS waas responsible for the mechanical subsystems, which consisted of the chassis, motors, and mountings. Team leads acted as SMEs for their subsystems.

Team leads reported to the chief engineer, who acted as the technical advisor and lead systems engineer. The chief engineer reported to the project manager. This role was responsible for budgeting, timelines, and the system's meeting of requirements.

2.2 Systems engineering model

GeoRGE applied the V-model method to their systems engineering, as seen in ***Figure 2***.

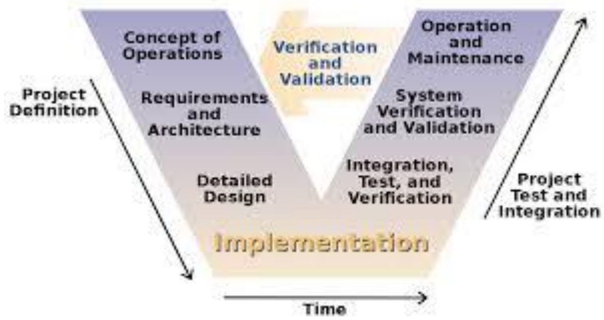

Figure 2. V-Model Diagram

There were major reviews at the end of the requirements and detailed design phases, as well as the beginning of the integration phase. These were the Systems Requirements Review (SRR), Critical Design Review (CDR), and Deep Dive Review (DDR), respectively. The DDR was added due to both teams being behind schedule of completion of their design, as well as the lack of preparedness seen at CDR from both teams. This review was intended to give teams time to ask for questions and and advice from the panel of judges.

3 Intended Versus Day-Of Operations

3.1 CONOPS

The Concept of Operations can be see in ***Figure 3.*** The stages have been detailed in GeoRGE CONOPS Description

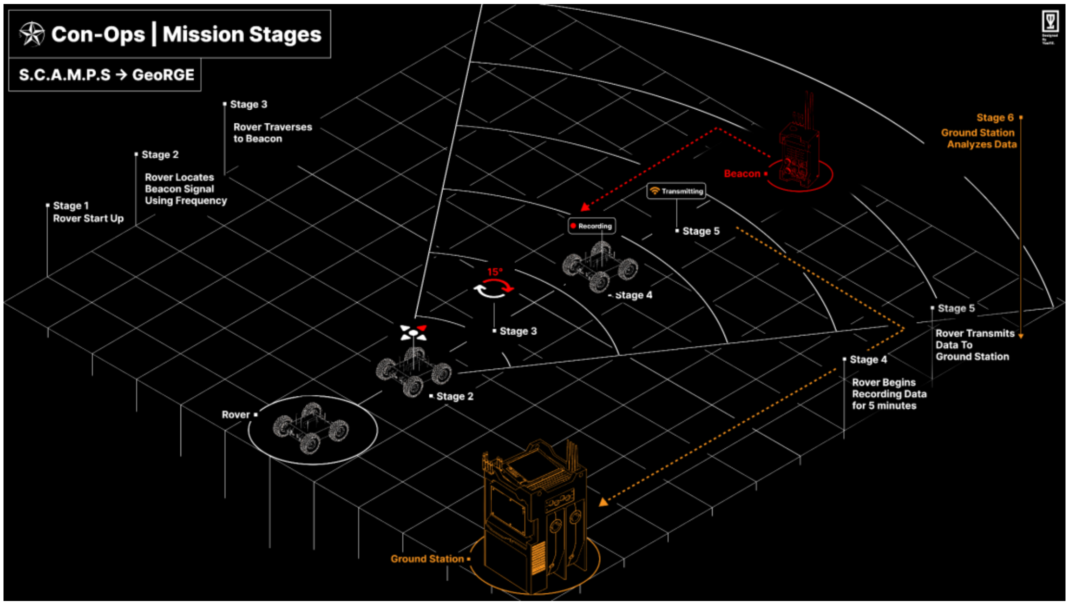

Figure 3. Concept of Operations

3.2 Comparison

The GeoRGE system followed an RVM that defined requirements of the system and subsystem to aid in determining what was necessary to have a successful operation. This can be seen in detail in GeoRGE RVM [3]

At the system level, the rover had 11 requirements, which are as follows:

- SR.1- The system shall be capable of operating for 1 hour.
- SR.2- The rover's length, width, and height shall each not exceed 40 cm.
- SR.3- The rover's mass shall not exceed 10 kg.
- SR.4- The cost of the system [rover and ground station sans laptop] shall not exceed $1,000.
- SR.5- The rover shall operate autonomously.
- SR.6- The rover shall collect environmental condition data for at least five minutes.
- SR.7- The rover shall transmit the data to the ground station.
- SR.8- The system shall analyze environmental condition data.
- SR.9- The rover shall locate the transmitting beacon in relation to the rover's current location.
- SR.10- The rover shall autonomously navigate to the beacon while avoiding obstacles in the terrain.
- SR.11- The rover shall operate within operating temperatures of all hardware.

These requirements were intended to be met successfully through the subsystem-level requirements. As seen in the GeoRGE RVM documentation, each subsystem requirement is linked to a system requirement.

The state machine diagram, seen in ***Figure 4***, displays the planned states of the mission. The states are further designed in GeoRGE State Machine Breakdown [4].

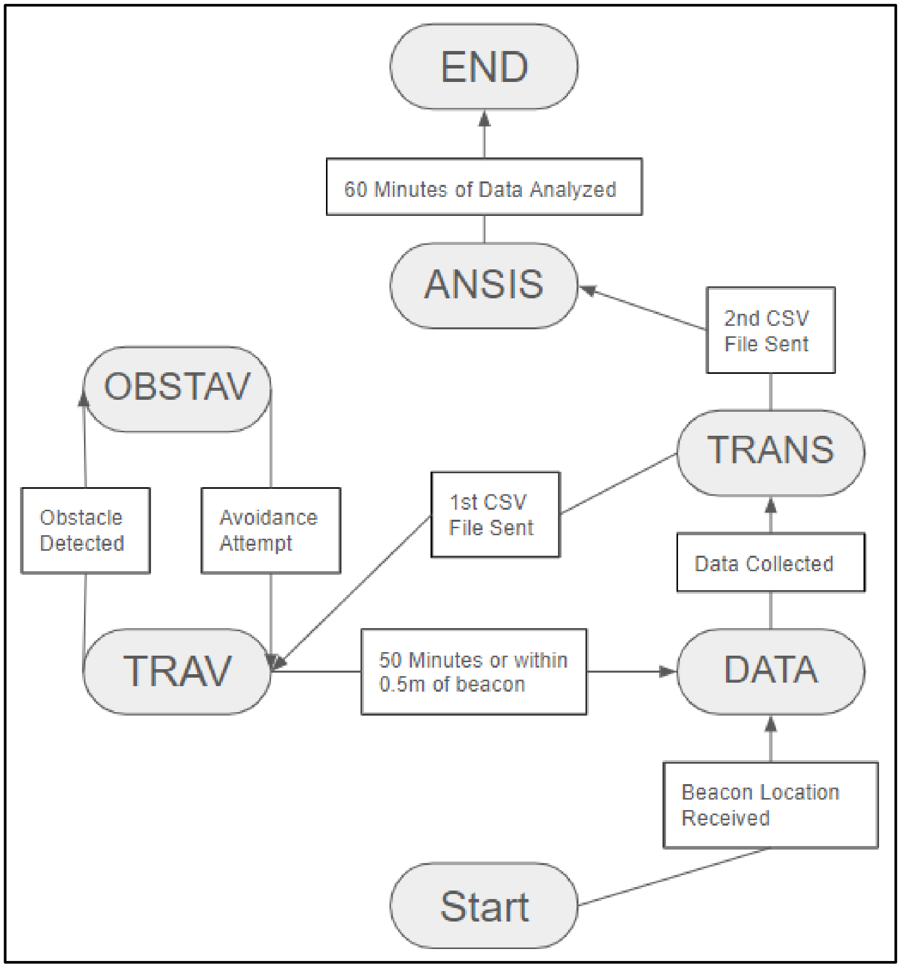

Figure 4. State Machine Diagram

The state machine diagram, the RVM, and CONOPS were used to determine the expected operations for the day of the competition. These, along with the actual day-of operation, can be seen in ***Table 1***. Actual operation outcomes highlighted in red are operations that failed. Actual operation outcomes highlighted in blue are operations that can not be determined a pass or fail, because they relied on the completion of failed operations.

Table 1. Expected vs Actual System Operations

Expected Operation	Actual Operation
Rover Startup	
Expected Operation	**Actual Operation**
Turns on and establishes a connection to GS	Turns on
Power is sent to antennas, Arduino, web cameras, Nvidia, and motors	Power is sent to antennas, Arduino, web cameras, Nvidia, and motors
Monopole antennas move	Deploys monopole
Serial connection established between Arduino and Jetson	Serial connection established between Arduino and Jetson
Backup Data Collection	
Expected Operation	**Actual Operation**
Sensors turn on initialize	Sensors initialized
Records data for 5 min	Did not record
Transmits raw data	Did not transmit
Rover Traversal of Terrain	
Expected Operation	**Actual Operation**
Locates beacon	Did not locate beacon

If obstacle, rover switches into OBSTAV state and avoids obstacles properly	Did not get a chance to locate and avoid obstacles
Rover Data Collection	
Expected Operation	**Actual Operation**
Sensors turn on	Did not get to start sensor collection at the beacon
Records data for 5 min	Did not get to record at the data site
Transmits raw data	Did not get to transmit from the data site
Ground Station Data Analysis	
Expected Operation	**Actual Operation**
Calculates average and standard deviation of each data set	Did not get data sent to calculate
Calculates density	Did not get data sent to calculate
Saves raw and averages	Did not get data to save
Shutdown	
Expected Operation	**Actual Operation**
Rover shuts down	Did not shut down after completion

3.2.2 CDH
The CDH subsystem had six requirements, which are as follows:

- CDHR.1- The CDH subsystem shall detect and categorize obstacles.
- CDHR.2- The CDH subsystem shall determine an optimized path to the beacon that avoids potential hazards.
- CDHR.3- The CDH subsystem shall receive sensor data from the EPS subsystem.
- CDHR.4- The CDH subsystem shall analyze experimental data.
- CDHR.5- The CDH subsystem shall receive beacon information from the COM subsystem.
- CDHR.6- The CDH subsystem shall determine the location of the beacon relative to the rover's current location.

The CDH subsystem successfully met CDHR 2, 3, and 4. However, CDHR 1, 5, and 6 were failed requirements. Though it was working prior to the competition, the CDH subteam made a last-minute decision to comment out the rover's ability to detect and categorize obstacles due to a realization that keeping it could cause the rover's software to completely crash. The same issue was determined for CDHR 5 and 6, though these were not tested to the extent that the first requirement was prior to the competition.

3.2.3 COM
The COM subsystem had four requirements (noted as COMR), and the antenna had an additional two (noted as COMA). The requirements are as follows:

- COMR.1- The COM subsystem shall detect the frequency of the radio signals from the beacon.
- COMR.2- The COM subsystem shall filter noise from the signals given off from the beacon.
- COMR.3- The COM subsystem shall receive sensor data from CDH subsystem.
- COMR.4- The COM subsystem shall transmit sensor data to the ground station.
- COMA.1- The Dipole Antenna shall detect signals emanated from the beacon.
- COMA.2- The Wifi Anenna shall connect to ground station and transmit data.

The COM subsystem successfully met COMR.4, however failed the other 5 requirements. Because of poor time management and last-minute component changes, many of these requirements were unable to be fully tested, and, although the rover was seemingly able to find a beacon near the end of the evening before the showdown, no true test was completed to verify this. Additionally, the rover's ability to do this was varying and was not able to be seen on the day of the showdown.

3.2.4 EPS
The EPS subsystem had three requirements, and they are as follows:

- EPSR.1- The EPS subsystem shall provide enough power for the rover to operate.

- EPSR.2- The EPS subsystem shall store enough energy for the rover to operate for one hour.
- EPSR.3- The EPS subsystem shall record sensor data.

Prior to the competition, all of these were successfully met. However, during the day-of operations, the requirements were not able to be demonstrated. Though the rover did turn on, the mission was not completed and, therefore, the subsystem never ran out of energy. The team was also unable to determine if the subsystem recorded sensor data because the system was only designed to be able to see the data on Ground Station.

3.2.5 TMS

The TMS subsystem had requirements (noted as TMSR), and the traversal unit had an additional two (noted as TMST). The requirements are as follows:

- TMSR.1- TMS subsystem shall be capable of traversing various terrains.
- TMSR.2- The TMS subsytem shall receive commands from the CDH subsystem.
- TMSR.3- TMS subsystem shall be designed within a structural factor of safety of 1.4
- TMSR.4- TMS subsystem shall be designed to house all components.
- TMST.1- The motors shall work unanimously with all other motors on the same side.
- TMST.2- The wheels shall be capable of spinning while subjected to the full weight of the rover.

The TMS subsystem successfully met TMSR.4 and TMST.2; It also partially successfully met TMSR.2. During testing, the rover's motors ran when and how the CDH subsystem told them to. However the subsystem failed to meet TMSR.1, 3, and TMST.1. The rover was not fully tested on its capabilities to run traverse the terrain. Leading up to the competition, one of the motor controllers was fried and the subteam had to pivot to only using four motors. Therefore, the rover could not produce the torque needed to traverse properly. Additionally, due to planning issues with the University's testing laboratories, the team could not fully confirm that the subsystem was designed with the designated safety factor.

4 Cause of Failure

4.1 CDH

4.1.1 Too Many Expectations for One Team

At the very start of the project the CDH lead made the decision to combine the CDH and GNC team into one and kept the name as CDH. Realistically a better name for the subteam would have been "SOFTWARE". The reason they made this decision was because at the start of the project there were 4 people on the whole of GeoRGE who were interested in programming and there were not enough people to split the team into GNC and CDH. As time went on there were consistently a limited number of people who were interred in coding and the scale of what all had to be done still hadn't sunk in. The subteam underestimated the difficulty of coding a proper ground station and the difficulty of a data transmission system. This led to the whole CDH team being spread thin. To mitigate this problem in the future, SCAMPS teams should make sure to split into GNC+CDH and ensure enough people are on board to fill those roles.

4.1.2 Poor Documentation

The original ICD diagram for the BBB, seen in ***Figure 5,*** displayed how other components would connect to the beagle bone blue, but it was not thorough enough to detail how it would connect. The original IDC diagram didn't detail which pins, which connectors were needed, and it never went into detail about how the code would interact with whatever data was being connected between other subsystems.

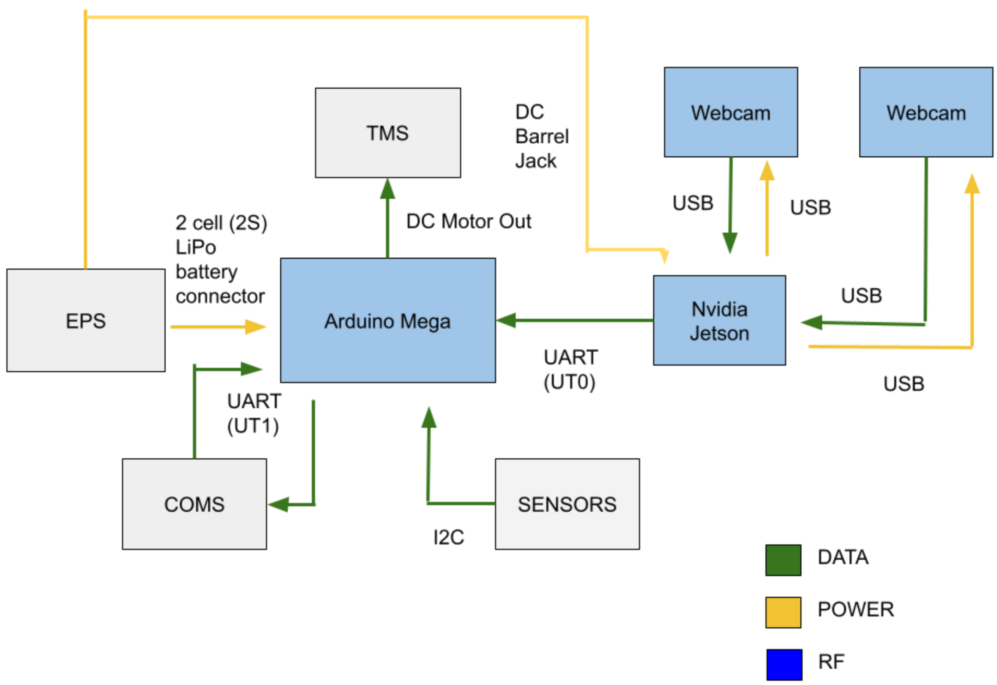

Figure 5. ICD for CDH Subsystem

Additionally, when switching from the BBB to the Arduino, the subteam never made another ICD diagram. Any connection information previously created wasn't centralized and was confusing to look at. This lead to two major problems on integration day:

- It took a long time to get all the motors to move because team members confused the pins that the motors were connected to and the wrong pins were in the code
- The NooELEC Radio module was unable to integrate with both the arduino and Jetson and we didn't know this until integration day. If a more thorough ICD diagram was written for the new system, this problem would have been caught earlier.

4.1.3 Internal Communication

The ML model and the stereo vision units worked separately, but the subteam never discussed what would be returned from the ML model, leading to the stereovision unit working under incorrect assumptions. By the time the stereovision worked in a way that would properly integrate with the model, it was too late to properly test the subsystem. The team should have asked more questions about the data being returned. Additionally, more conversations should have been had as the subcomponents were being created to ensure all members were on the same page. This would have allowed for less errors and an earlier integration timeline. Finally, the team did not create a spreadsheet with parameters the code will need and what will return. This would have greatly helped the skeleton code phase.

4.1.4 Technological Issues

There were multiple technological issues seen in the CDH team's components. As one example, the BBB was incredibly difficult to work with, to the point where the team had to change micro computers. This occurred because of improper trade studies that put too much emphasis on cost rather than component familiarity, wide spread use, community support, and component support from the creators. Additionally, the subteam waited too long beore interfacing with the BBB. Unfortunately, the difficulties that were seen immediately when trying to put the ArduPilot on the software were not given enough time to be resolved because of this.

The board was also quite difficult to flash images onto, which led the team to thinkt he board was accidentally corrupted for a while. Once the image was flashed on the BBB and the team could establish a shared network protocol, the team tried, but failed, to connect the board to the Mission Planner software. Specifically, the problem seen was 'Unable to connect UDP port to 14550.' The following troubleshooting steps were taken:

- Checking the port for alternate traffic
- Restarting Mission Planner
- Restarting the computer
- Reinstalling Mission Planner

- Checking network drivers for updates, disabling and enabling relevant drivers
- Attempting the installation on three different computers
- Reflashing the image on the BBB

The team restarted the entire setup process on three different computers, but the same error always occurred. Once they had tried everything, there were only six weeks until competition, and the team chose to switch to the arduino platform, which has a much larger support platform and more documentation.

Another technological error that was run into was the Nvidia Jetson being old. To cut costs, the team used an old Nvidia Jetson donated to the team by a member. Though the Nvidia Jetson was in working condition, it had not been updated to the current version. Updating the Jetson took a lot of time away from matters that also needed attention.

4.2 COM

4.2.1 Poor Time Management

One major problem the COM team encountered near the end of the project was the lack of time needed for adaptation to problems surfacing towards system level integration. As will be further expanded on in Section 4.2.2: Poor Documentation, the most important unit, the NooELEC Radio module failed to be integrated into the rover and thus eliminated the RF finding subsystem. With the integration start date, the COM subsystem was wholly unprepared to adapt to the new demands of the rover and time quickly ran out.

4.2.2 Poor Documentation and Research

Poor documentation and research was one of the main reasons why the COM subsystems failed to be integrated with the other subteams. The Ground Station failed to satisfy the range needed to meet the mission requirement due to its short range. In addition, the antenna system was not fully researched and its limitations were finally understood right before the competition, which was too late to make changes to plans. Overall, constant research and development should have been conducted through the duration of the project and, most importantly, communicated to all team members on the project.

4.2.3 Poor Communication

Throughout the development of the SCAMPS, poor communication was a constant issue seen during the project. We faced issues with integration previously stated and we believe the lack of communication among the subteams led to the overall failure of integration. With the replacement of the BeagleBone Blue with the Arduino master board, the COM subsystem failed to sufficiently communicate and adapt to the new board. It was not until system integration that it was discovered that two of the system's units, the Arduino board and the NooELEC Radio Module, could not be interfaced with one another.

4.2.4 NooELEC Radio Module

During the system level integration of the radio module to the rover, the subteam encountered an issue where there was no module or documentation online for communication between the radio module to the arduino. The arduino could not interpret the information being received from the radio module nor could the radio module initialize its device for listening. The radio module, however, did communicate with the JetsonNano via the USB port. The main reason the team had to use the arduino for communication was the replacement of the BeagleBone Blue, discussed in 4.1.4, thus the subteam was unable to utilize the USB ports for communication. The NooELEC radio module is used best via Python and the accompanying RTL-SDR package.

4.2.5 Lack of Antenna Testing

With the delayed start of system level integration and system level testing, the COM antenna system was never properly tested and configured. Through subsystem testing of the system, both the direction finding test and the distance detection test were successful but were designed for the BBB and, therefore, could not be properly integrated with the arduino. When the rover system was fully integrated there was not enough time for proper integration and was eventually scrapped to focus on other mission-essential functions.

4.2.6 Short Range of WiFi

Initial consideration of utilizing WiFi was heavily influenced by familiarity with Transmission Control Protocol (TCP) between two systems. WiFi is useful for wireless communication but failed to be useful in long distance communication. The module we used was the ESP32 Development Board and used a 2.4GHz WiFi chip for wireless communication. It was intended to work as seen in ***Figure 6***.

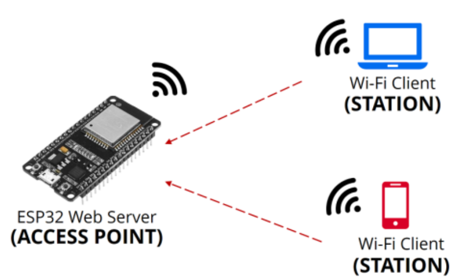

Figure 6. Diagram of Intended WiFi Communication Setup

Testing was conducted and it was determined that the maximum range of the WiFi subsystem was around 130 meters, which was significantly less than the mission requirement of 354 meters. The ESP32 was easy to program using the ArduinoIDE but was not the correct choice for the COM ground station system.

4.2.7 Lack of Ground Station Testing

The testing of the ground station was conducted 48 hours before the competition started. The ground station was constructed, coded and tested within this timeframe and was operational before mission start. This short time frame did not give the COM subteam ample time to work on other subsystems such as the antenna system on the rover. This issue resulted from and contributed to the overall lack of time management experienced during the duration of the project.

4.3 EPS

4.3.1 Motor Drivers

The motor drivers that were used for GeoRGE was the TB6612FNG. Each motor driver could control two separate motors. Since GeoRGE was a tank drive with 6 wheels, 3 motor drivers were used.

During testing of each motor driver individually, they were able to successfully control each of the two motors attached to it. But once all 3 motor drivers were tested all together to the power supply and arduino, the team placed the controllers with exposed copper onto the aluminum frame. Additionally, they did not check the power supply being used to test before turning it on, resulting in them running 30V through the controllers. This fried all three boards. More had to be ordered last minute. For the final batch of 3 motor drivers, one motor driver worked perfectly fine, the second motor driver had the inconsistent problem, and the third one just fully stopped working. For the second motor driver, one motor would work, while the opposite motor would not. This would keep happening after checking if it was a wiring issue, a connection issue, a motor issue, a code issue, a pin issue, and a power issue, but all of those were triple checked and were fine.

The issue was isolated to the motor driver itself, because when measuring the voltage across the output pins to the motors that wouldn't spin, the voltage read by the multimeter was 0. Whereas the voltage across the output pins to the motors that did spin, had a value. One possible issue was heating. Not much proper testing to fully state this as a heating issue, but the reason for this is still unknown. The setup for all motor drivers were exactly the same, so there's not enough evidence

Regardless of the specific issue, possible reasons for causing this problem that weren't tested/confirmed include

4.3.2 Servo Feedback

The servo feedback to obtain the current angle of the servo was tested by itself and it was confirmed to work. During integration though when mounting on the rover with all other electronics, the angle that was measured was very inconsistent and would not stay constant when held at a constant position. One possible reason for this issue that wasn't taken into consideration was that during integration, power was supplied through a 5V regulator instead of the arduino during testing. The reason for using the 5V regulator was because the 5V pin on the Arduino isn't able to supply the max current, so it has to pull current from an external source. Both grounds of the 5V regulator and the Arduino also had to be connected, which also wasn't tested. Overall, there was not enough testing to ensure it would work with the entire system iteratively.

4.3.3 Poor communication in selection of Microcontroller/Development Board

The microcontroller for the rover was a major component that was used by all subsystems (CDH, EPS, COM, TMS). The selection of this component should be discussed with all subsystems before finalization. At the beginning of this project, CDH had selected the BeagleBone Blue for their subsystem and EPS had designed their system to be integrated with CDH, COM, TMS via the Beaglebone Blue.

As integration and testing progressed, there were unforeseen issues with CDH's use with the BeagleBone Blue and GeoRGE switched to using the Arduino Mega. This lead EPS to having to redesign connections and integration with CDH only a month before competition.

If EPS had worked with CDH earlier and performed testing and integration in advance, this problem would have been foreseen and mitigated quicker.

4.3.4 Wire Organization

The organization of the wires, or lack thereof, as seen in ***Figure** 7*, was part of the reason integration took as long as it did. Wires would get mixed up, causing troubleshooting to take longer as well. A few wires fell out or got lost and had to be replaced and re-soldered. Overall, wire organization could have helped a lot in resolving issues a lot quicker had it been done. A lot of issues came with integration, and the wiring only added to that.

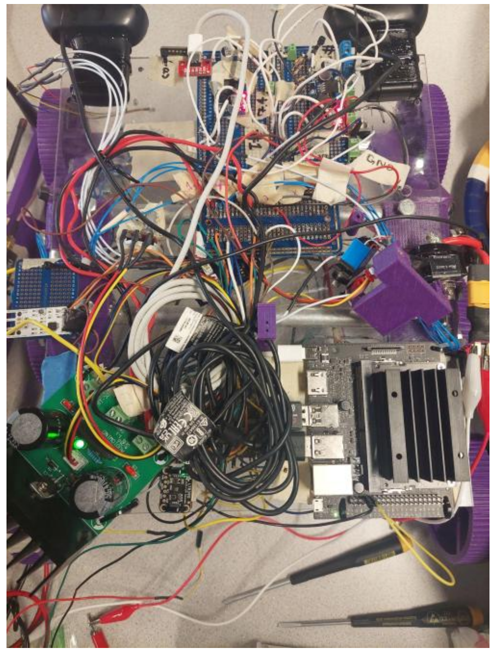

Figure 7. Example of Wire Organization Issues

4.3.5 Late Integration

Given the multitude of challenges faced with motor drivers, wire organization, microcontroller, and servos, significant strides toward inter-team integration were deferred until the latter stages of the design process. Despite receiving extra time for testing, the bulk of it was consumed by ongoing issue resolution. Prior to the EPS subsystem's seamless integration with other subsystems, it was imperative to first ensure the functionality of its own components. However, persistent setbacks with existing devices confused the available time for integration, as rectifying EPS-related issues took precedence. A more strategic approach to mitigate these challenges would have entailed proactive anticipation and prevention of potential issues, enabling early collaboration with CDH and TMS teams. Such early engagement would have helped significantly. A good example is particularly in coordinating wire management with TMS, as this could have saved time through ensuring optimal space utilization.

4.4 TMS

4.4.1 Pivot From Six to Four Motor System

The rocker-bogie, seen in the SolidWorks model in ***Figure 8***, was designed to act as both a chassis and suspension for a vehicle. Due to this, it is necessary for the vehicle to operate utilizing a six wheel drive system. The intention of utilizing so many motors allows for the rocker-bogie system to traverse and travel over obstacles bigger than the wheel diameter. For this to happen, each set of wheels need to be able to drive a majority of the vehicle's weight independently.

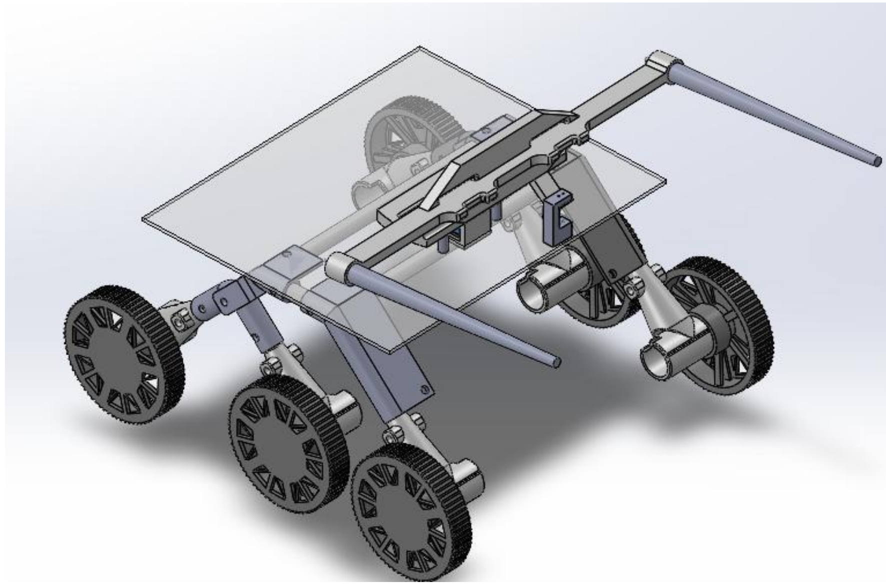

Figure 8. SolidWorks Model of TMS Subsystem

Due to the issues outlined in 4.2.1, the team had to pivot from using six motors to four. The switch was done the week of the competition, leaving no time to redesign the drive system to accommodate only four wheel drive, additionally negating the purpose and requirements for a rocker-bogie. Instead of having six driving wheels, the front two and back two wheels were driving while the middle wheels were being driven.

4.4.2 Electronics Mounting Issues

The TMS subsystem team did not coordinate with the other subsystem teams about where their respective electronics needed to be mounted and housed until the last few weeks of the competition. This lack of coordination had the most impact on EPS. Because the teams did not take the time necessary to figure out the wiring required for their electronics to mount onto the chassis, EPS spent much more time soldering and desoldering to make all the components fut than necessary. Additionally, due to last second decisions about wiring and lack of updated documentation, the end product was visually a mess and difficult to work with and around, as displayed in ***Figure 9***. Many wires were fully exposed to the elements and there was no wire management set in place. This set a high risk of wires being ripped from boards or getting caught on the ground during traversal. If decisions had been made earlier, TMS could have designed a method to conceal and protect the electronics better.

Figure 9. Completed Rover

4.4.3 Designing for Manufacturability and Assemblability

During the design phase, the TMS subteam utilized SolidWorks to design many of their 3D-printed parts. However, little thought was put into the tolerancing between the aluminum of the frame and the 3D prints. This led to very tight fits or improper fits which had to be manually filed to make work. This also caused testing to be pushed back due to the time it took to fix these errors. The subteam also neglected to consider assemblability of the design. This led to issues in the alignment of the wheels, making the system incredibly difficult to drive straight.

4.5 System

4.5.1 Lack of Awareness on System Engineering Method

Due to many changes in the lab over the last few years with little communication to or between project teams, many people lost who was in charge of onboarding. The onboarding process involved teaching new members about the V-Model, which is the typical model used in the laboratory. This used to be a programmatic responsibility, but was shifted to the ShadowCorps. However, not all members go through the ShadowCorps before entering a project team. The higher leadership joined when programmatics handled it and, therefore, did not train new members on the V-Model. This led to many errors in the systems engineering of the project.

Because subteams did not understand the V-Model or system's engineering, there were lots of issues with individual ownership instead of working towards the common goal. Each subteam had an idea of what the rover was meant to be and meant to do. Though there was an RVM that helped guide the overall idea, less conversations were had past the creation of that document, which led to a breakdown of uniformity. Upper management did not do their job in catching this until it was pretty far in the designated desige phase timeline. This significantly pushed the design phase timeline back

Another issue was that team members did not understand the need for testing individual units and integrating in small steps, testing along each checkpoint. Some tests were completed, but it was not prioritized in the way it should have been. This led to many tests not being completed, which caused a ton of integration issues.

4.5.2 Documentation Update Lapses

Upper management did not prioritize updating documentation as plans changed. At the end, there were many design, testing, integration, and assembly changes that had occurred without proper documentation. This made it very difficult to give instructions on the final days leading up to the competition because there was not an official set plan that had been approved and written down. Best practices were not shared with others and documented. Errors and fixes were not written down during processes. At the end, attempts were made to fix old documentation, but so many changes had been made that there was not enough time to get everything down and many changes were forgotten or missed in documents.

4.6 Human Error

4.6.1 Team Member Departures and Onboardings

Throughout the project, there were many members that left. Many of these members left with little to no warning. This led to a lot of loss of institutional knowledge, as upper management was unable to ask members to write down their learnings and processes for those that would come to replace them. Additionally, because of the necessity to move quickly and the small size of the team, new members were not given adequate time to learn the previous decisions made on the rover and the design of the system. They were forced to jump in and be up to a similar speed to members that had been in the laboratory for years. This led to newer members progressively being more confused on the overall purpose and mission.

4.6.2 Communication Lapses

As mentioned in 4.5.1, there was a lot of individual ownership of subsystems. This was also caused by communication lapses. Though there were subteam lead meetings, they did not occur often enough due to members not being able to find a time that worked for routine meetings and an overall lack of preparedness for meetings by not only team leads, but also upper management. Though the project manager prepared questions for these meetings, they were not sent out in advance, which did not give subteam leads time to think about answers that would have given other leads a better picture of the working of the rover. Additionally, team leads oftentimes did not come prepared with questions for other subteams. Finally, meetings were mostly only called during the weekly team meetings when there was an obvious issue, with an occasional check in subteam meeting. This led to lapses in the overall vision until there was a catastrophe nearby multiple times. The subteam leads should have had meetings with the project manager and chief engineer either weekly or biweekly to make sure all subteams were following the same vision.

5 Future Recommendations

Though there were many problems seen through the programmatic and technical delivery of the system, leading to an overall unsuccessful mission, the team was able to identify many recommendations for future members to follow.

5.1 Communication Between Subteams

In the beginning of the project, there was a lot of collaboration and communication between subteams. Leads and upper management worked together to define requirements that aided in completing the overall mission. However, as the team transitioned to the design period, this communication fell off, which led to issues on subteam and system levels. Future teams should ensure communication between subteams at all stages of the process. They should also have weekly team lead meetings re-establish the overall mission and make sure all subteams are up-to-date on any changes that are being made so they can evaluate where these changes impact design decisions made.

5.2 Timeline Accountability

Though upper management did create a Gantt Chart, they had trouble enforcing team members to stick to the established timeline. They also did not update and account for issues that arose that pushed back major milestones set in place. Future teams should find ways to strongly enforce timelines and instill the belief that the timeline is not very flexible among members. They should also frequently check and update the timeline as changes are made to the system and issues arise.

5.3 Documentation Updates

Documentation errors accounted for a large portion of issues seen through the competition. In the future, team members should be introduced to and trained on documents before being expected to fill them out properly. For example, trade studies are highly effective when done properly. However, not all members were fully aware of how to prioritize their constraints in trade studies, leading to poor component selection decisions and unreliable boards. Additionally, upper management and team leads need to work together to consistently update all living documents as changes are made. This will ensure that all members are able to find proper knowledge and have an official, set standard for work they will be completing.

5.4 Official Training

It is highly recommended that trainings and onboardings are set up at not only at the team level, but at the program level as well. Members should have required trainings on the programmatics of the organization, similar to that of a company onboarding. These trainings should include overall expectations, safety, team overviews, and what models and systems are followed overall. On a team level, further training should be provided about how the projects are designed, current best practices of the role they are taking over, and introductions to all subteams. The implementation of these two trainings would greatly benefit both new members and current members on teams.

Additionally, people who can be identified as experts in certain subjects should work to teach others in the organization about best practices. This could be done either in module creation or workshops. These subjects could include, but are not limited to, testing procedure creation, SolidWorks design for manufacturing, soldering, and PCB design. There is a lot of institutional knowledge among members, but there is not a system in place to ensure that knowledge is not lost upon members leaving or graduating.

6 Conclusion

Though many mistakes were made that led to a failed mission in many regards, the lessons learned were incredibly invaluable. When looking at the core of the laboratory, this is the true goal. Members received knowledge of a beginning to end-of-life product lifecycle and, although numerous errors were made along the way, this experience still helped prepare them for successful careers through instilling knowledge of following proper and improper practices. Through documenting the process and these lessons, future members can push themselves further, learning from mistakes they have the privilege of not having to make.

7 References

[1] Scamps 2023 Overview
[2] GeoRGE CONOPS Description
[3] GeoRGE RVM
[4] GeoRGE State Machine Breakdown