5-2019

Design of a Model Rocket Flight Logging System and In-Air Deployable Rover

Sirus Negahban
Union College - Schenectady, NY

Design of a Model Rocket Flight Logging System and In-Air Deployable Rover

Sirus Negahban

ECE-498 Computer Engineering Capstone Design

Advisor : John Spinelli

UNION COLLEGE

FOUNDED 1795

CONTENTS

List of Figures

LIST OF TABLES

Abstract—**The goal of the project laid out in this paper is to develop a model rocket range and altitude tracking system and a payload for said rocket which conducts an experiment of some scientific merit. The requirements for the project are defined by the rules of an international model rocket design and build competition for undergraduate and graduate students. This paper presents a design to accomplish the specified goals to the standards of the competition rule set, for use by the Union College Rocket Team at the competition. First, an off-the-shelf flight computer is purchased to implement the range and altitude tracking requirements. The selected component is researched thoroughly and tested to ensure it performs as expected. Next, a design is developed for a scientific payload known as the Habitability Assessing Research Vehicle, or HARVe for short. HARVe is a rover which is stored within a model rocket and, depending on the rocket's capabilities, can be deployed while in flight or on the ground after the rocket has landed. Once it is free of the rocket and on the ground, it will roam the landing zone and assess the habitability of the area through imaging and air quality tests. As of the submission of this report, HARVe has been prototyped to complete a number of its final functions, yet testing and further implementation of these abilities continues as the design is iteratively improved.**

Index Terms—**altitude logging, Experimental Sounding Rocket Association, flight computer, habitability, model rocket, payload, rover, Spaceport America Cup, Union College Rocket Team**

I. Introduction

The Union Rocket Team is an engineering design team founded in 2017 with the purpose of competing in an international model rocketry competition hosted annually by the Experimental Sounding Rocket Association (ESRA). The 2018-2019 Rocket Team is composed of 3 Senior engineering students, namely Daniel Brack '19, and Madeleine Miller '19, and myself. The team is advised by Professor Andrew Rapoff and Professor John Spinelli. My role on the team is primarily concerned with the design and implementation of all on board electronics required by the the competition and any additional electrical functionality desired by the team. Although I am responsible for those parts in particular as the sole computer engineer, the team heavily encourages interdisciplinary cooperation, and my teammates and advisors have offered and given me help as readily as my teammates have invited my input in their own design challenges. Thus, although each member of the team is responsible for his or her own part, these three Capstone design projects will form a singular final product which belongs equally to all members.

ESRA asks that participating collegiate teams choose a target apogee of either 10,000 or 30,000 ft design and build a model rocket to get as close to the chosen height as possible. This offers an opportunity for students from all over to see through an engineering challenge motivated by competition and interest in aerospace. The Union College Rocket Team has decided to compete within the 10,000 foot bracket this year. ESRA also offers a more open ended design challenge as a part

of the competition. Rockets must be loaded with not insignificant “boiler-plate“ weight to act as a ballast during launch, and although teams are able to submit a rocket design with such a payload, ESRA encourages teams not stop there. A payload which replaces the pure ballast weights and achieves a technical or scientific objective which makes good use of a 10 or 30 thousand ft apogee flight will be given a significant bonus to scoring and the potential to win a cash prize. In the spirit of the challenge, The Union College Rocket Team has decided to design a rover to be either deployed from the rocket during descent or on the ground after landing and capable of independently collecting data which will provide insight into how the suitable the landing zone is for living. The rover has been named Habitability Assessing Research Vehicle, or HARVe for short.

The goal of this project are to design an electronics package for the Union College competition rocket which meets the qualifications defined by ESRA and the team. Additionally, this project is also concerned with ensuring that HARVe lives up to its name, and is able to explore and evaluate the habitability of an area independent of any of the systems on board the rocket. In the report to follow, design choices will be made for how best to select a flight computer and integrate it into the rocket, in addition to structural and electrical design choices for HARVe.

II. Background

Aerospace engineering has seen many leaps in progress in the last decade unheard of since the start of the space race over 60 years ago. In the US, the National Aeronautics and Space Administration (NASA) was born in a relatively unique period of history during which time the agency received almost unending support from the US government. While this enabled its meteoric growth, it also left the agency completely unprepared for an existence without this support[1]. For several years, NASA would live in the shadow of its golden age, and many programs would have to be cut for the agency to survive. Then, in 2004, privatized space travel was legalized under the Commercial Space Launch Amendments Act (CSLAA)[2]. No longer hidden behind bureaucracy and defense contracts, companies like Boeing and Ball Aerospace who had been filling contracts for NASA for decades now have more incentive than ever to invest in future aerospace engineers. New companies such as SpaceX and Virgin Galactic seem to be growing even faster than NASA first did, and the more excitement generated about space exploration the more successful each company will be. Even though private space companies are competitors in terms of services offered, they share a common goal of hoping to intrigue people with the field of aerospace engineering.

A. The Intercollegiate Rocket Engineering Competition

The birth of the commercial space industry brought with it a need for a new generation of aerospace engineers and technicians. Rocketry and space exploration has an important place in the classroom, and many organizations have been created with the express purpose of transforming that classroom experience into something more tangible. ESRA is one such organization. It was started in 2003, and has since gained recognition with its Intercollegiate Rocket Engineering Competition (IREC). Competitors from universities and colleges around the world aim to launch a rocket closest to a target apogee of either 10,000 or 30,000 ft. The competition features many prominent sponsors from industry such as Boeing, Blue Origin, and Virgin Galactic, all private space exploration agencies looking to grow the profession of aerospace engineering. In this interest, students are given the opportunity to design a model rocket adhering to competition guidelines. From the challenge provided by ESRA, this interdisciplinary capstone design project was designed to enhance further the already sophisticated standards set out, and push forward the spirit of cooperation which is vital to the future of space exploration.

The Union College Rocket team will be entering its second year as a competitor in the ESRA Spaceport America (SA) Cup. The rules as defined by the IREC guidelines have strict requirements so far as the primary electronics package to be carried by the rocket, however what is delivered above and beyond these requirements is entirely up to the design team. Teams are encouraged to outfit their rockets with scientific or experimental payloads as opposed to simple ballast weight, but the choice is ultimately up to the students. Teams are also motivated to adopt real industry standards in their rocket, such as the CubeSat, defined as a volume of 10 x 10 x 10 cm^3 and used to most efficiently allocate cargo space within a rocket. These more technically complex challenges heavily impact manufacturability of the rocket, but provide a huge benefit to the usability. Having a scientific payload adds value to the mission, which now has an objective beyond reaching a target apogee. Using the CubeSat standard enables the rocket to guarantee cargo space for any other potential cargo. Designing a rocket flight system and technical payload in accordance with these standards offers many new practical design challenges which a team of students may not otherwise experience.

In addition to the physical challenges presented in manufacturing, there are also legal barriers which cause increased difficulty for the project. What makes the SA Cup such a unique opportunity is the altitudes to which the rockets are expected to go and what additional considerations such heights necessitate. Primarily, it is illegal in the US to launch high

powered rockets with a total impulse of up to 40,960 N-s without explicit authorization from the FAA[3]. Current designs for Union College's competition rocket, to compete in the 10,000 ft bracket, require a motor with an impulse of approximately 7656 N-s. Thus, the rocket team is incapable of launching a full sized rocket until the competition, as there are no model rocketry clubs near Union College which have the same accreditation and approval as ESRA to launch such a rocket. Thus, it may prove difficult to know the full range of conditions the on-board electronics and payload will be exposed to until launch day at competition. Additionally, the dangers of shipping fragile or volatile components means that the project will need to be completed far enough in advance of competition to ensure

III. Design Requirements

A. IREC Required Components

Many design requirements are dictated by the rules of the SA Cup, but several others are necessitated by the team's own desired functions. Thus, building the rocket is as much an exercise in specification and precision as it is creativity. The former provides the structure upon which all colleges will actually be competing, and from this competition dozens of unique designs for rocket and payload functionality will be shared and celebrated. The competition design requirements put forth by ESRA as related to this capstone design project are primarily concerned with rocket recovery and data logging. According to the official SA Cup IREC Rules[4], the criteria for a passing electronics package on a rocket include,

1) A Comercial Off The Shelf (COTS) barometric pressure altimeter with on board data storage.
2) A transmitter capable of communicating location information via GPS or equivalent global navigation satellite systems.
3) Optionally, any telemetry systems must only transmit data that originated from the same sensor source as the on board data log.

The rules require a COTS barometric altimeter to ensure that the accuracy of the data is not the responsibility of each team, but instead guaranteed by a manufacturer. A transmitter capable of communicating the rockets location is required for quick recovery of the rocket body after landing. Although telemetry systems are optional, including one can greatly streamline data recovery and adds a layer of redundancy should any systems on board the rocket break, and is therefore desired by the team. Furthermore, as per the Rocket Team's preference, the rocket electronics system should be on a physical switch which is accessible even after the electronics have been sealed within the rocket. With these rules in mind, additional consideration was given to the design requirements of electronics system of the project with input from Union College Rocket Team

members and advisors.

B. Parachute Deloyment

The on-board electronics must also be capable of deploying parachutes during descent, and based on the recommendations of previous year's teams and competitors the rocket should have 2 parachutes, known as a dual deploy system. These 2 parachutes will be a smaller one called the drogue deployed near apogee and a larger parachute called the main deployed near the ground. The function of the drogue is to slow down the rocket as it descends, however maintaining enough speed so that it has enough downwards momentum to counteract the force of wind on the parachute. This principle is demonstrated in Figure 1, which compares the ground distance drifted from the peak height of a rocket flight for dual and single deployment systems.

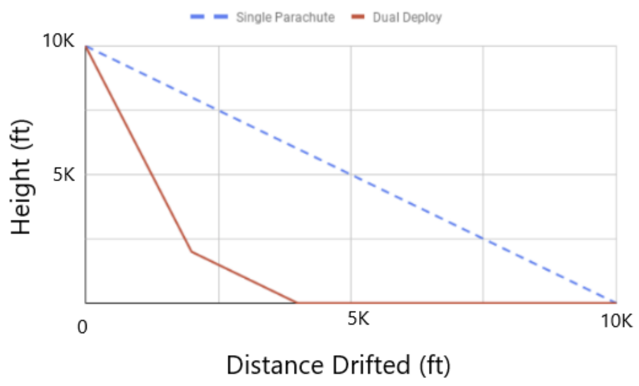


Fig. 1: Descent trajectories of rockets containing dual and single deployment systems.

Figure 1 shows the benefits of the dual deploy system, as the recovery is made much easier by the dual deployment system. Since this method is preferred by the Rocket Team, it adds an additional requirement to the on board functionality of the electronics, namely that it be able to deploy 2 parachutes at different times and as a function of distance from the ground. The responsibilities of the electronics are not to perform the mechanical decoupling of the rocket body to release the parachute, but rather to simply send the signal at the correct time. The rocket body will be split by a black powder charge ignited within the sealed rocket, creating just enough pressure to push the halves apart.

C. Radio Communication

The primary design requirement of all radio systems on board the rocket is that they be able to communicate with the rocket at every point in the flight. The competition is held in a desert in Truth or Consequences, New Mexico, so there will not be many obstructions between the ground station and the transmitter on the rocket. Thus, it can be assumed that much of the communication between the two will occur via line of sight propagation of the radio waves. If significant bending or penetration of objects were required of the radio waves, ultra or very low frequency communication would be required, which is usually restricted to government or military use. The ability of these waves to bend is due to the Huygens Fresnel principle, which dictates that waves propagating at lower frequencies will diffract more markedly behind obstacles

than those of higher frequencies[6]. Line of sight propagation is typical of devices with much higher transmission frequencies, as these wavelengths may not bend around obstacles as well but can do so sufficiently when there are only minor obstructions. Some higher frequency radio waves are available for civilian and commercial use, and this range of frequencies is referred to as the Industrial, Scientific and Medical (ISM) band[5]. To understand more of what will be required from the on board radio, range calculations are necessary.

Although the rocket will be designed to achieve an apogee of 10,000 ft, this distance may not be an accurate reflection of the maximum range required of the radio transmitter. The rocket motor will be carefully sized to propel the known weight of the rocket, so the margin of error for the maximum height of the rocket is not nearly as wide as the potential drift distance of the rocket after parachute deployment. A worst case scenario for the drift of the rocket would involve the main parachute deploying at apogee. If this were to happen, then an approximate drift distance $d_{drifted}$ based on the descent rate $v_{descent}$ of the rocket from last year's team[7], average wind speeds at altitude in Truth or Consequences v_{wind} [8], and the height of the rocket h can be obtained.

$$d_{drifted} = \frac{h}{v_{descent}} * v_{wind}$$

$$d_{drifted} = \frac{10,000}{18} * 36.96$$

$$d_{drifted} = 20,533.33 \text{ ft.}$$

Although it is unlikely that the main chute would deploy with the drogue at apogee, it is necessary that the radio continue working in this scenario for recovery, making the maximum range approximately 20,533 ft. The ability of a radio to transmit such a distance depends on the output power of the transmitter and the antenna capture area of the receiver. As a wave propagates through space, the power density of the signal in a given area adheres to the inverse square law, meaning that doubling the range requires 4 times as much transmission power for the receiver to read the same signal with the same power. Figure 2 [9] demonstrates this phenomenon.

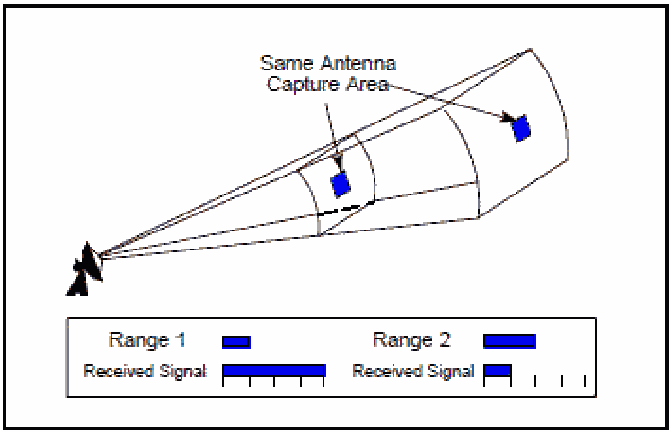


Fig. 2: The power density of a radio signal as it travels.

Thus, a suitable radio for the rocket is a line of sight frequency transmitter not defined its ability to transmit from the ground to apogee, but from the the ground to a worst case scenario drift distance of the rocket, the latter requiring approximately 4

times the power as the former.

D. Flight Computer Powering

In addition to specifying the performance of the sensors on board the flight computer, the team is also concerned with ensuring the flight computer as is powered properly. Flights at competition last an average of 2 minutes, and recovery of the rockets takes an average of 15 minutes [7]. All of the electronics should have a battery capable of powering them for such a duration to a reasonable factor of safety, say at minimum 30 minutes.

The team must be capable of manually switching all rocket electronics on or off externally from the rocket as per IREC Rules [4], meaning a switch must be mounted on the of outside the rocket body. The switch chosen for this task must be very robust, as it will be entirely exposed to external forces during flight and on the ground, and its state must not be toggled unless intentional. The system does not need to be capable of remote start-up or shut-down.

E. Scientific Payload: HARVe

The IREC rules do not provide many constraints as to the design requirements of a scientific payload, other than it have a weight of at least 8.8 lbs or approximately 4 kg. Thus the payload is largely made to the agreed upon specifications of the Rocket Team and advisors. The general list of requirements for HARVe post-deployment is as follows.

1) Photograph landing zone during descent/after landing.
2) Absorb landing impact without damaging internal electronics.
3) If in-air deployment, decouple parachute from rover
4) Evaluate temperature, humidity, altitude, and Volatile Organic Compound (VOC) gases.
5) Store data on board and transmit via telemetry.
6) Explore landing zone in a spiral for a set amount of time.

Expanding on these functions, we can classify each into one of three categories. These are habitability assessment, data logging and transmission, and finally navigation and movement. Each of these sub-functions of HARVe is defined in greater detail in the sections to follow, in addition to structural requirements not constrained by rover function.

1) Structure: IREC rules indicate that additional points will be given to team's whose scientific payloads can be adapted to fit into a 3U CubeSat, which is equivalent to stacking 3 standard CubeSats on top of one and other. It is the desire of the team that HARVe adhere to this standard in the interest of achieving the best possible score. Thus, the outer dimensions of HARVe may not exceed 30 x 10 x 10 cm. As mentioned in the preceding section, according to IREC rules the overall weight of HARVe must be at minimum 8.8 lbs, with an

error of 0.4 lbs[4]. To minimize torque requirements of the motor, it is preferable that the wheels on the rover be as large as possible, meaning the diameter should be as close to 10 cm as possible, or approximately 9 - 9.9 cm or 3.5 - 3.9 in.

2) Habitability Assessment: This sub-function is composed of all tasks which require a sensor to capture data which may later be used to evaluate the habitability of the landing zone. This includes any aerial or ground photographs taken by HARVe, as well as measurements of temperature, humidity, pressure, and VOC gasses. A combination of humidity and VOC gas data will be used to determine local air quality. A sensor must also be chosen to detect deployment from the rocket, as this event will trigger all other functions of HARVe. Images from the photo-sensor require the greatest amount of storage space of all the sensors, so a low sampling rate over the descent time is desired so as not to run out of space. Resolution of the sensor is not particularly important, as the square area represented by a single pixel in the image will decrease as the sensor approaches the ground, providing a sort of natural zoom effect. Other sensors must be chosen such that they can be most easily interfaced between allowing for simple collection of data to be stored and transmitted. For example, altitude data must be stored on board the rover, and must be accessed by other components as a means of knowing when the rover has stopped descending.

3) Data Logging and Transmission: All data collected by HARVe must be stored on board the rover. In the event that some part of the rover is damaged, the on board storage should be removable so that data can be recovered independent of rover functionality. This on board storage system has similar requirements to the data-logging system on the actual rocket, meaning similar or potentially identical systems may be considered for both. Similarly, radio communication between the rover and the ground station will have near identical requirements as the rocket's radio transmission system.

4) Navigation and movement: If HARVe is deployed in-air, its descent will not be controlled, meaning that in both deployment scenarios there is no active navigation required before the rover reaches the ground. Consider the in-air deployment use case. The desired descent speed of the rover based on parachute sizing calculations from the Rocket Team is 5 m/s, and anticipating a deformation of about 1 cm of a padded landing surface, such as a rubber wheel, on which HARVe could land, the impact force of landing will be approximately

$$F = m * \frac{v^2}{s}$$

$$F = 4\text{kg} * \frac{(5\text{m/s}^2)^2}{0.01\text{m}}$$

$$F = 10,000\text{N}.$$

Thus, the rover, assuming it will land on its wheels, will have to absorb an impact force of

10,000 N. Once it has landed, HARVe will need to navigate in an exploratory spiral while continuously collecting data. HARVe has no specific speed requirement, so a suitable speed can be as low as 0.25 to 0.5 $\frac{\mathrm{m}}{\mathrm{s}}$. Torque may be prioritized over speed, as a worst case scenario requires the driving motors pull the 4 kg of weight against the full force of gravity, if it were scaling a small ledge for example.

With the goal of maximizing wheel diameter, preliminary calculations can be performed to size motors to drive HARVe. A single motor can be appropriately sized to drive the rover, meaning that any additional motors will add a sufficient factor of safety. Assuming relatively low friction between the wheels and ground, a given a wheel diameter d of 9 cm and a desired speed v of 0.5 $\frac{\mathrm{m}}{\mathrm{s}}$, the approximate no load speed ω of the motor is obtained as follows.

$$\omega = \frac{v}{\pi * d} * 60$$

$$\omega = \frac{0.5}{\pi * 0.09} * 60$$

$$\omega = 106.1 \text{ rpm}$$

While HARVe is moving at 0.5 $\frac{\mathrm{m}}{\mathrm{s}}$ on flat ground, the motor will have to overcome very little force to turn, specifically just the friction between the wheels and the ground. A desirable motor will then have a no load speed not too much larger than 106.1 rpm. In order to size the motor completely, an estimation of the stall torque must also be calculated. As mentioned, a worst case scenario for HARVe is when it must pull it's full weight over an obstacle. Again, using a wheel diameter of 9 cm and a the required weight m of 4 kg, the stall torque τ can be obtained as shown.

$$\tau = m * a * \frac{d}{2}$$

$$\tau = 4 * 9.8 * \frac{0.09}{2}$$

$$\tau = 1.76 \text{ N}$$

Thus, if HARVe is to lift itself over an obstacle, the driving motor must have a stall torque only slightly greater than 1.76 N or approximately 249.24 oz-in. With estimations for the no load speed and stall torque, an optimal motor can be described as a motor which operates near the estimated values within a certain error. Figure 3 is a plot of the estimated speed-torque equation of the desired motor with 20% error thresholds. The chosen motor may not perform to this threshold strictly, but should contain many points within the margin of error.

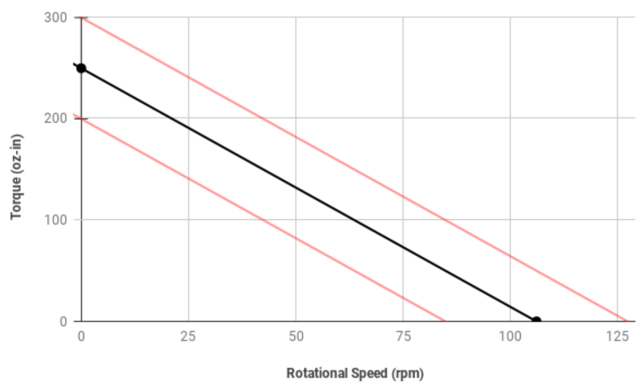


Fig. 3: A speed-torque plot for the desired motor given a 20% margin of error.

IV. Design Alternatives

A. Rocket Electronics Selection and Alternatives

Based on the design requirements described in the preceding sections, a top level diagram for the on board electronics was created and can be seen in Figure 4. The system is initiated by a single on/off switch, after which point the altimeter will set the current altitude as the zero height, and will then continuously output its height data to the input of the transmitter. The altimeter will also control the 2 output signals for parachute deployment, to be sent to the deployment mechanism. The GPS or equivalent module tracking location similarly sends its data to the radio transmitter, which then sends the information it's collected to the ground station.

IREC requires the altimeter be COTS for ease of data verification when examining the system for scoring. Many COTS altimeter options are offered by a variety of companies which complete various combinations of the tasks described in the design requirements. Last year's Rocket Team recommended the Telemetrum V2.0b[7], a flight computer which is capable of recording altitude, GPS location, temperature, in addition to having 2 output channels capable of dual deploy and a 434 MHz radio on the same board. The radio operates within an ISM band, and has a maximum power of 40 mW. To determine the range of the radio, the Friis transmission equation is used assuming an antenna gain, G, of -6 dBi. Transmission power P_{TX} in dBm, receiver sensitivity P_{RX} in dBm, and frequency f in MHz is given by the manufacturer datasheet for the particular radio chip on the Telemetrum[10]. Since the radio waves propagate by line of sight, the link margin L_M can be approximated to 0 dB. Thus, first the path loss L_{FS} is found, and then used in the Friis transmission equation to solve for range r as follows.

$$L_{FS} = 36.6 + 20\log f + 20\log r$$

$$L_{FS} = 89.33 + 20\log r$$

$$P_{RX} = P_{TX} + G_{TX} - L_M + G_{RX} - L_{FS}$$

...by substitution of L_{FS} and solving for r...

$$r = antilog(\frac{P_{TX} + G_{TX} + G_{RX} - P_{RX} - 89.33}{20})$$

$$r = antilog(\frac{16 - 6 - 6 + 109 - 89.33}{20})$$

$$r = 15.25 \text{ miles}$$

$$r = 80,520 \text{ ft}$$

Although 80,520 ft is only a rough approximation of the range of the Telemetrum's radio chip, it is significantly larger than the anticipated maximum distance between receiver and transmitter of 20,533 ft, and is thus within a considerable margin of error. According to the Telemtrum data sheet, it has been range tested by the manufacturer as far as 40,000 ft, and should work up to nearly 100,000 ft[11]. The Telemetrum V2.0b is a strong choice

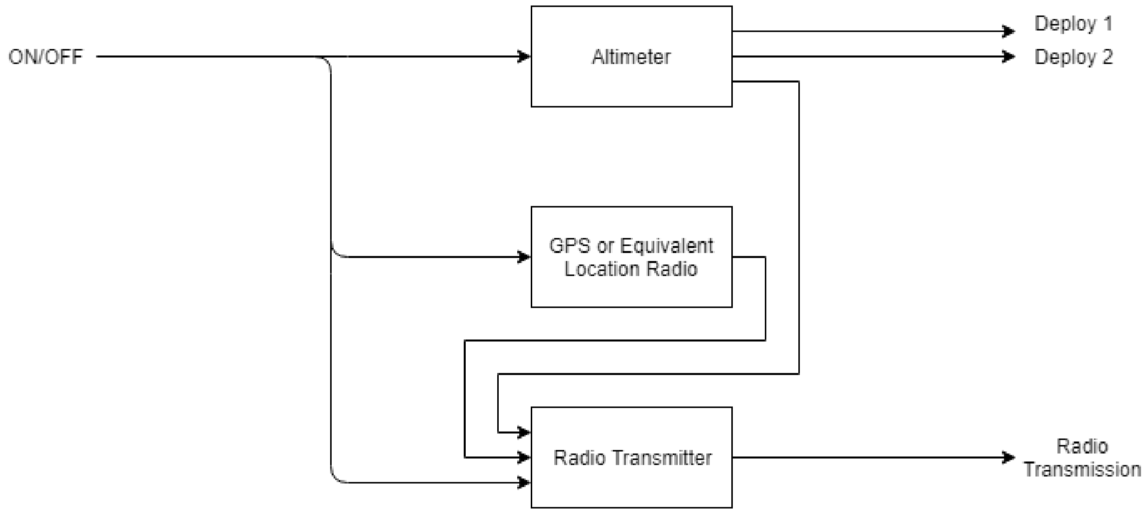


Fig. 4: Top level diagram of the electronics system within the rocket body.

for the rocket flight computer, as it incorporates all necessary functionality onto a single board, has an ISM band radio which is suitably powerful for the expected range, and is highly recommended by the previous year's team.

It is typical for teams competing in IREC to choose a system like the Telemtrum, where much of the functionality of the flight computer is contained in a single system. With this in mind, other flight computers such as the TeleMega and Perfectflite Firefly were considered. The TeleMega is a flight computer with all of the functionality of the Telemetrum, but it includes additional parachute deployment channels which could be used to eject the payload of the rocket independent of any other recovery events. With the Telemetrum's 2 channels, payload ejection has to occur simultaneously with either the drogue or main parachute deployment. However, purchasing a TeleMega flight computer would be an additional cost of over $400, and thus is not cost effective. The Perfectflite Firefly flight computer is significantly cheaper, at only $27, but does not include a radio transmitter or GPS module, and thus additional modules would have to be purchased and interfaced with the computer. Thus, the Telemetrum fulfills the IREC requirements as a COTS system and satisfies all additional design requirements as desired by the Rocket Team.

The Telemetrum includes a monopole whip antenna, which is suitable for the range of nearly 100,000 ft quoted by the manufacturer[11]. The rocket body is primarily constructed of composite materials which are radio transparent, so the transmitter should maintain good line of sight with the receiver over the duration of the flight. That said, more work should be done to consider alternate antenna shapes, as the full range cannot be tested before competition, so it is desirable to select an antenna which will ensure as strong a transmission signal as possible during flight.

B. HARVe Structure Configuration and Alternatives

Although HARVe is constrained by the 3U CubeSat standard in width, height, and depth, it's overall shape is nearly entirely unconstrained. Based on motor sizing, it is preferable to fix the wheel diameter to approximately 9.9 cm. For optimal stability, the wheels should be as far apart as from each other as possible, which would seat them approximately 30 cm apart with the rover structure enclosing the axis between them.

Four wheeled design alternatives were considered, however using four wheels significantly reduces the available space for the rover structure, and adds a significant degree of difficulty when turning, as without steering all four wheels must rotate at different speeds to follow different arcs on the turn.

On the 2 wheeled axis 3 different rover structure shapes were considered. The first was a simple cylindrical body seated between the two wheels, with a cross section shaped as a circle. The second is a structure in the shape of a rectangular prism seated between the two wheels, the cross section being a square. The third and final design considered is a hybrid shape, where the cross section is a circle on the bottom half and a square above, and this shape is extruded and sits between the wheels. To demonstrate what each shape may look like, Figure 5 displays a cross section of each of the rover structures imposed over wheels of equal diameter. In order for each rover body to have the same tolerance for ground obstructions, the minimum distance from the set containing all points on the wheel and its nearest point on the bottom half of the rover body must be the same for all structure shapes. This rule was used to scale the cross sections in Figure 5 to the fixed diameter of the wheel. The structures are evaluated in the decision matrix in Table 1 by the categories of volumetric efficiency (Vol.), ease of manufacturing (Man.), and structural integrity (Str.).

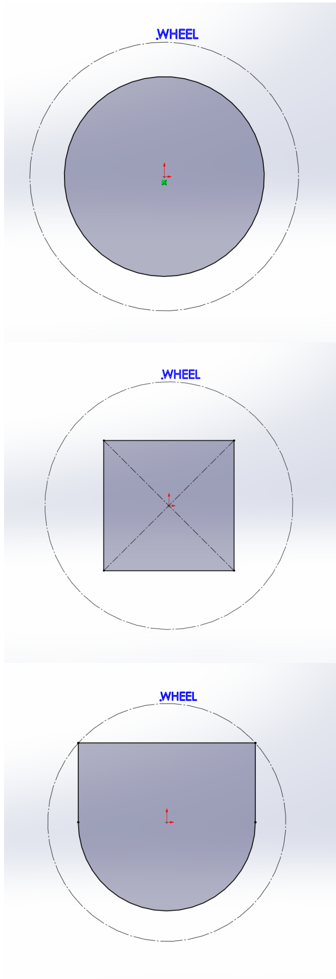


Fig. 5: Cross section of the cylindrical rover body (top), rectangular rover body (middle), and hybrid body (bottom).

The decision matrix demonstrates that a cylindrical rover body is the ideal shape. It scores

	Weighting	Circular	Square	Hybrid
Vol.	0.3	4	2	5
Man.	0.5	4	4	2
Str.	0.2	3	2	3
Total		**3.8**	3	3.1

TABLE I: Decision matrix for HARVe structure shape. Each category is scored on a scale from 1-5.

second highest in volumetric efficiency as it will necessarily always be smaller than the hybrid body and larger than the rectangular body. The cylindrical structure is similar in manufacturing difficulty to the rectangular body, but both are much easier to build than the hybrid body. Finally, it ties again in terms of strength, this time with the hybrid body. Corners are structural weak points on the body, however realistically only the bottom half of the structure has the potential to become deformed due to this weakness on landing or when going over an obstacle. Thus the rectangular rover body scores lowest in terms of strength. Knowing the properties of each of the potential body shapes for HARVe logically leads to the decision to use a cylindrical shape for the structure.

Using a two-wheeled design may grant better usage of space, however it comes at the cost of stability. HARVe is likely to roll around the central axis as it moves, as there will only be 2 points of contact with the ground. Thus a skid should be deployed upon landing, which will protrude out from the rover and rest on the ground in front of it. This will provide a third point of contact for stability and prevents HARVe from rolling about its central axis.

C. HARVe Electronics Design and Alternatives

Since HARVe must run a single program to complete a predetermined series of functions with known sensors, the system is ideal for an embedded micro-controller. Although there are several chips available which is capable of interfacing between the kinds of digital/analog sensors required for the habitability measurements, an Arduino Uno is chosen because of the team's experience with the Integrated Development Environment (IDE). Additionally, the Uno provides pin headers so debugging and testing of each component will be greatly simplified. The Uno contains an 8-bit micro-controller based on RISC architecture. Knowing that an Arduino Uno will be used, sensors to complete other rover functions will be chosen to easily interface with the Uno, of which there are a wide variety.

V. Preliminary Design

The subsections of the preliminary design section to follow represent the design work done on the project in its early stages. The final design has remained, for the most part, faithful to the parameters laid out here. To elaborate on the changes made to the final design, section VI has been included in the report to reflect the most recent iteration of the project design.

A. On Board Flight Computer Design

The electronics on board the rocket are used primarily for altitude tracking, location tracking, data logging, and parachute deployment. If any one of these functions fail then in accordance with IREC rules the launch will not receive a score and is removed from consideration in the competition. The secondary function is to transmit logged data from the rocket to a ground station in real time, however scoring is not dependent on having working telemetry.

1) Flight Computer System Design: The functions described above and in the design requirements section are all contained within the Telemetrum V2.0b flight computer. Figure 6 is a complete wiring schematic for the flight computer as it will be contained within the rocket body.

The battery used to power the flight computer is a single cell LiPo battery with an accepted output of 3.7 V and a capacity of 900 mAh. The battery was supplied with the Telemetrum, and additional design requirements for battery sizing were not considered because the maximum current consumption of the computer is 150 mA[11], at which rate the 900 mAh battery would still power the system for about 6 hours. Peak current consumption only occurs while attempting to lock a GPS signal, thus the supplied battery is well oversized for the flight computer. The battery weighs only 0.85 oz, meaning it accounts for less than 0.15% of the total liftoff weight of 40 lbs, and any weight saved by choosing a smaller or more appropriately sized battery would be completely insignificant.

The flight computer is mechanically isolated from the battery by a single-pole, double-throw (SPDT) switch. The reasoning for this is because any number of conditions may cause the assembled rocket to sit at the launch pad for an extended period of time prior to launch. As per the design requirements of the system, the switch will be accessible from the outside of the rocket. Finally, the Telemetrum's 2 pyro channels will be used to detonate the charges responsible for decoupling the rocket during descent.

2) Flight Computer Event Progression: The sequence of events of the undergone by the flight computer is very particular, prematurely triggering any phase could completely ruin the flight and serious damage the rocket. In order to understand how the computer executes this sequence, the source code running on the flight computer is examined. The first of the phases identified by the computer is a stable landing pad phase, which begins when the Telemetrum is turned on in launch mode. During this phase the computer will operate in a low power mode, where the sampling rate is reduced to 10 Hz as opposed to the typical 100 Hz sampling rate during flight. If the rocket is upright and experiences a particular acceleration, the altimeter will exit low power mode and begin the boost phase. This is the

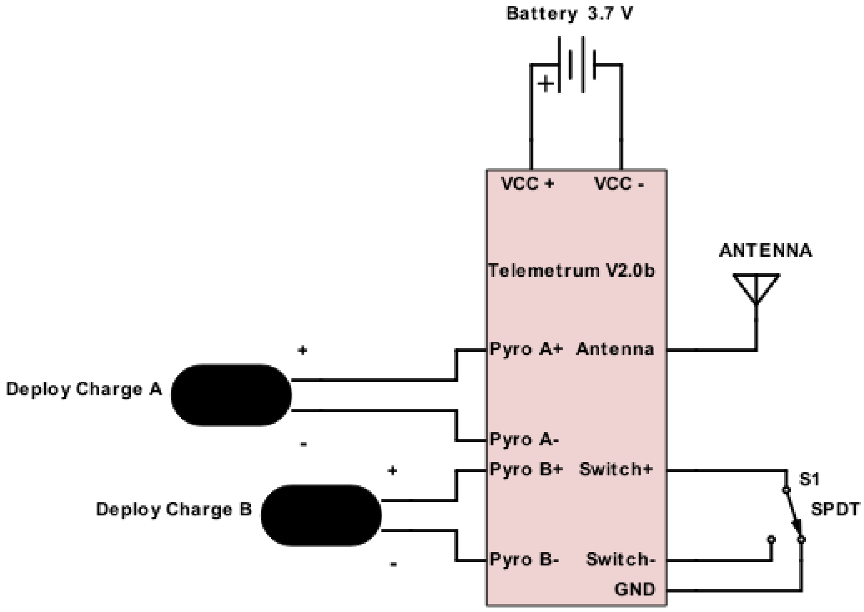


Fig. 6: Schematic of the electronics system within the rocket body.

phase during which the rocket is powered. After the motor has expelled all of its fuel, the altimeter begins the coast phase, where the rocket is traveling up un-powered. Once the rocket reaches apogee, the drogue parachute is deployed as per IREC rules[4], thus beginning the drogue phase. It is upon entering the drogue phase that the first parachute, the drogue parachute, is deployed. IREC rules also state the main parachute in a dual deploy system should not deploy above 1,500 ft, so once the the rocket passes this threshold the computer should begin the main phase, at which point the main parachute will be deployed. Finally, after landing solidly on the ground the altimeter will return to it low power, stable phase.

Figure 7 depicts the sequence of events as a state diagram, and each input is quantified and defined in Table II following. The Telemetrum uses the on board altimeter and accelerometer to progress the sequence of events, typically based on whether or not a particular threshold is achieved.

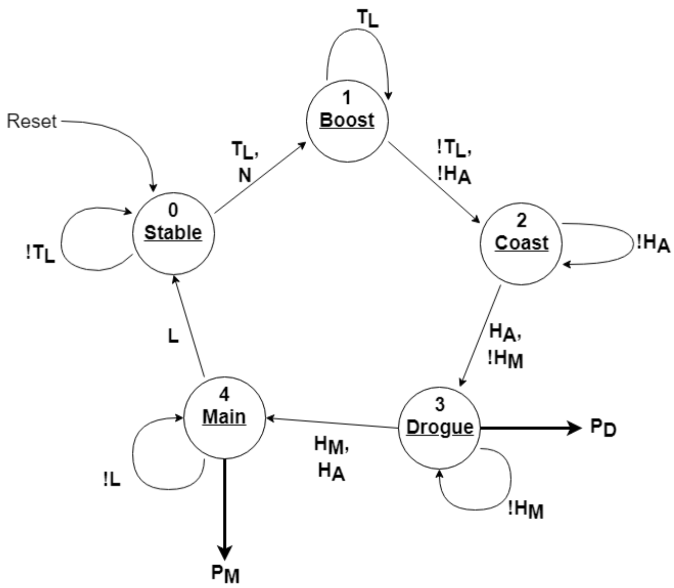


Fig. 7: State diagram demonstrating the sequence of events as executed by the flight computer.

The Telemetrum often checks whether or not

Symbol	Sensor Reading
T_L	Launch trigger, 2 G accel. and 5 $\mathrm{m/s}$
N	Accelerometer detects upright position
H_A	Apogee height reached during flight
H_M	Altimeter reads below 1,500 feet
P_D	Signal drogue parachute deployment
P_M	Signal main parachute deployment
L	End of descent and 0 G accel.

TABLE II: Definitions of input and output signals used in Figure 7

apogee has been reached when changing phase. Prior to apogee the computer is concerned only with tracking the current phase of the rocket. Post apogee the computer is also responsible for outputting a signal to deploy the parachutes at set altitudes, however an instantaneous height collected during rocket ascent is identical to an equivalent height collected during rocket descent. To avoid premature parachute deployment the computer will only send deployment signals if it has also recorded an apogee, ensuring that parachutes will only launch if the rocket is descending. Future testing may reveal ways to in which the team wishes to change the behavior of the Telemetrum, and knowing how the source code functions and compiles will make it possible to alter the functionality to the team's specifications.

B. HARVe Design

The performance requirements of HARVe have been enumerated in previous sections, however chief among them is its ability withstand landing impact and navigate terrain at the landing zone. HARVe's sensors to evaluate habitability provide a unique programming and interfacing problem, however the motors driving the rover have join the worlds of mechanics and circuitry. Thus they are subject to both electrical and physical problems, and will need to be chosen to withstand both. Beginning at the top level, Table III is a functional decomposition of the rover and its parts. The final design for the sensors and motors on board HARVe are presented in the sections to follow, but first the design of the containing structure will be detailed.

Function	Component
Detect deployment from rocket	Photoresistor
Photograph terrain during descent	OV5642 5MP Camera
Detect landing	Altitude from Adafruit BME680
Deploy skid and decouple parachute	Servo motor
Explore landing zone in a spiral	DC Motors
Evaluate temperature, humidity, Pressure, and VOC Gas content	Adafruit BME680
Transmit data to ground station	HX1 - VHF Narrow Band FM APRS Transmitter
Log data	Adafruit MicroSD card breakout board

TABLE III: Functional Decomposition of HARVe

1) HARVe - Structural Design: In accordance with the design decisions justified in this paper, the shape of the rover will be a cylindrical tube suspended between 2 wheels, such that the distance between the outer rims of the wheels will not exceed 30 cm and the diameter of the wheels will not

exceed 10 cm. A skid will extend from the rover body to provide a third point of contact with the ground for increased stability. Figure 8 is a crude rendering of what such a structural design might actually look like.

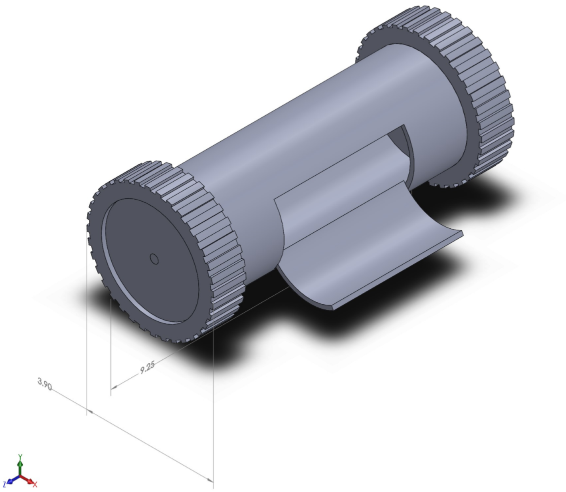


Fig. 8: Initial CAD model of HARVe.

The cylindrical body of HARVe has a diameter 1 in. less than the wheel diameter, meaning the rover should be able to pass over obstructions 0.5 in. tall without much difficulty. Rather than store an appendage within the rover to be extended as the skid, HARVe is designed to allow for rotation of a rectangular cutout of its outer structure on a hinge which will be locked in position by a servo motor and used to enhance stability.

After approximating the structure of HARVe, more formal design work of the rover was done with the help of the Rocket Team. According to IREC rules, HARVe must weigh at least 8.8 lbs or 4 kg. The contribution of the electronics towards this goal is relatively negligible, save for the motors and battery. The majority of the required weight of the rover would need to be in the actual structure, and with this in mind the team decided to construct as much of the rover from steel as possible. With manufacturability and weight goals in mind, the team redesigned the rover to be composed of 6 major pieces. Figure 9 displays such design for HARVe, with the 6 pieces labeled and elaborated on in Table IV following.

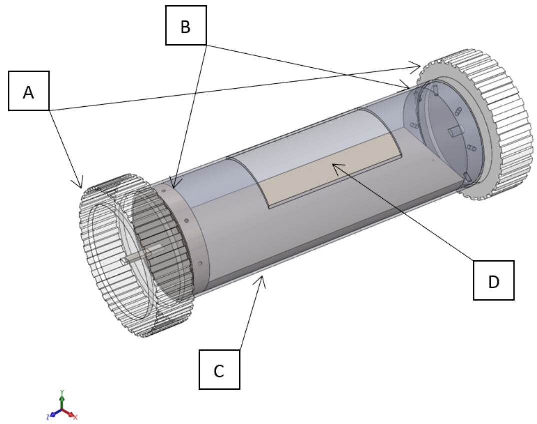


Fig. 9: Refined CAD model of HARVe, Note transparency applied to left wheel, main body tube, and right body tube end cap. For annotations see Table IV.

Many decisions were made about the structure of the rover with the requirement in mind that it may potentially have to withstand 10,000 N or more upon impact. Rubber wheels will deform to reduce the impact as much as possible. The half inch thick steel end caps provide the structure for the rover, but also protect the coupling between the wheels and the rover body as well as the motors which will be mounted within. Despite this, the majority of the impact force from any drop will be dispersed across

Part	Description
A	Rubber wheels, designed to keep main body tube approximately 0.5 off level ground at all times.
B	Steel body tube end caps. DC motors are mounted to these, and drive shafts will protrude from rover body and attach to the wheels.
C	Steel main body tube. This piece will contain the electronics used onboard the rover.
D	Steel electronics mounting plate. This piece sits flush to the inner radius of the main body tube, and has flat surface on top for electronics mounting. Also serves as a ballast weight to increase stability while driving.

TABLE IV: Descriptions of labeled pieces in Figure 9

the 2 DC motor shafts attached to the wheels, which is another factor to consider when sizing motors.

2) HARVe - Motor Sizing: Based on the calculations from earlier selections, the Pololu 12 V 131:1 DC brushless geared motor was was selected to drive the rover. To understand why this motor is an appropriate selection, it is helpful first to look at Figure 10, where its own torque vs speed plot is imposed over the estimated plot generated earlier in Figure 3.

With a no load speed of 80 rpm and a stall torque of 250 oz-in, the chosen 131:1 motor is suitably within the design constraints previously defined. Two of these motors should have very little issues pulling the 8.8 lb rover up hills and over rocky terrain. Of course, the shafts of these motors will be completely exposed to the full force of impact

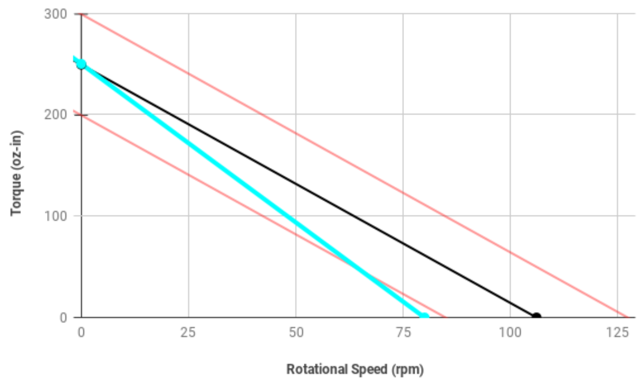


Fig. 10: Torque vs speed of the selected 131:1 motor (blue) imposed over the desired torque vs speed curve.

upon landing. A worse case scenario would see the rover land at an angle, causing all of the 10,000 N to be taken on by a single motor shaft. To understand what kind of damage such a crash might do, the pressure exerted on the steel motor shaft. According to the manufacturer, the motor has a shaft diameter of 6mm. Knowing this we can obtain the pressure.

$$\gamma = \frac{F}{\pi * (d/2)^2}$$

$$\gamma = \frac{10000}{\pi * (0.006/2)^2}$$

$$\gamma = 353677651.3 \text{ Pa} = 353.68 \text{ MPa}$$

Steel has a worst case ultimate yield strength of approximately 400 MPa[12], which is just above the worst case scenario pressure experienced by the motor shaft, 353.68 MPa. Thus, although the shaft likely will not break entirely in a worst case landing, it is possible the shaft could deform, still leaving the rover unable to move about. Potentially, a collar could be used to provide some additional rigidity

and strength to the motor shaft. The extent to which the shaft might deform is hard to know without undergoing testing, but needs to be considered to ensure the rover is able to land safely.

The motors requires a separate battery capable of outputting 12 V to operate. For such high voltages, it is desirable to use LiPo batteries because they are the most volumetrically efficient, however weight is not a design constraint. With this in mind, it is desired that HARVe be able to explore for 10 minutes continuously, and since no other components will be using this battery it is okay if it is entirely depleted. Assuming maximum current I draw of 5 A from the motors during this time, a suitable battery capacity C to run 2 motors can be calculated as follows.

$$C = 2 * I * \frac{10}{60}$$

$$C = 2 * 5000 * \frac{10}{60}$$

$$C = 1,666.67 \text{ mAh}$$

Although a battery with a capacity of nearly 1,700 mAh is quite large, and it is doubtful that the chosen motors will draw peak current for 10 minutes continuously, any weight and volume taken from the battery by choosing a smaller capacity will have to be made up with ballast weight to ensure HARVe achieves the weight requirement. Thus, it is preferable that the additional volume and weight be useful to the mission, and so a larger battery is preferred.

3) HARVe - Electronics: Use of an Arduino Uno microcontroller greatly simplifies sensor selection and communication, as many sensors are made for the microcontroller, and it contains enough pins such that the entire electronics array can be interfaced with a single Arduino Uno. To see a complete schematic of all the components which will be carried on board the rover see Appendix A, and it is recommended that this be used as a reference for the design outline to follow.

The 131:1 geared motors are driven by the MC33926 Motor Driver, which is capable of powering 2 motors at 12 V each while supplying a continuous current draw of 3 A and a burst current of 5 A. The path of the rover as it explores will be a spiral of constant radial increase, which the motors can navigate while being driven at a constant speed. Therefore it is not particularly important that motor speed be tracked by an encoder, however with this functionality a simple PID controller could be implemented to guarantee a that the rover does not speed up or slow down over its path. For now though, encoders have been omitted from the final design due to a lack of available pins on the Arduino Uno. Thus, the motor speed will be blindly controlled by a PWM signal from the Arduino uno.

The radio chosen for the rover is the Radiometrix HX1 VHF transmitter, an ISM band radio capable of communicating over a maximum range of approximately 32,000 feet [13]. The HX1 operates

on a regulated 5 V power supply, making ideal for use with the Uno. The HX1 has one pin capable of reading in data to transmit, and data will be passed to it through the Uno as it is collected by the various sensors.

In addition to data transmission, all measurements will be stored on board HARVe via flash storage onto a micro SD card. Using a micro SD card allows for quick access of data on another system should something be damaged on board HARVe. Flash storage is also extremely resistant to physical damage because there are no moving parts, so it is ideal for data logging on the rover. Data is sent to an Adafruit micro SD card breakout board from the Uno via serial peripheral interface (SPI) communication, where it is then written to flash storage.

The optical sensor on board HARVe also communicates with the Uno via SPI, which is used as a two way communication channel where the camera receives commands from the Uno and sends images over a data-stream. The Arducam OV5642 Camera Module was chosen partially because its usage of the SPI channels makes it easy to communicate with, but also because it has an astonishingly high sensor resolution of 5 Megapixels, which, although still lower resolution than any given smartphone camera, is much better than alternative image sensors available with Arduino libraries. As explained in the design requirements, the camera will be able to resolve increasingly finer areas as it approaches the ground, but the additional image resolution certainly will help at higher altitudes. Being a 5 Megapixel sensor, it will require suitable space in which it can save images, which will be provided by the flash storage. The camera also utilizes the Arduino's I2C interface for configuration of the image sensor.

The camera will share the I2C interface with the Adafruit BME680, which is the combined temperature, humidity, barometric, and VOC gas sensor. The BME680 will send measurements from all of these sensors to the Uno over the I2C interface, however it also has another important function on board the rover. The barometric sensor is accurate enough to calculate altitude via a simple function to within an error of ± 1 meter according to the datasheet. Thus, the barometer can be used to signal landing to the Uno after HARVe's ejection from the rocket. Realistically, the accuracy of any altitude measurements provided by the BME680 is not so important as the ability of the sensor to detect that the change in altitude is 0 or near 0, as this will be a better indication of landing anyway.

The final 2 components in the schematic in Appendix A are the photoresistor and servo motor. The photoresistor is wired up as a simple voltage divider with a 220 Ω resistor, however so long as the impedance is a known quantity the resistor could have any value. Analog pin 0 on the Arduino Uno

will read in a voltage value, for which the Uno will then be able to calculate a brightness. Since the rover will be sealed within the rocket until ejection, the Uno will be able to identify when the rover has decoupled from the rocket based on voltage values at pin 0 surpassing a certain brightness threshold. At this point, the Uno can begin requesting image data from the camera module, and the can begin functioning as desired. Once the rover has landed and confirmation of this is given by the BME680, the servo will extend the skid for increased stability in movement.

VI. Final Design and Implementation

A. Motivation For Design Updates

The project presented in this report represents a only a component of a larger project which is the rocket being built by the Union College Rocket Team as a whole. Thus, as the needs of the team change, so too do the requirements of the project presented here. This section has been added to the report to explain what changes have been made in the months since section V, preliminary design, was written. The original content of the preliminary design section was preserved because, in addition to largely still being accurate, it provides context to the portions of the design updated below.

B. Flight Computer Final Design

The Telemetrum V 2.0b flight computer will be used in the final design in the same capacity as was detailed in the preliminary design. The complete board can be seen in Figure 11, which also details the layout of the major components on the board.

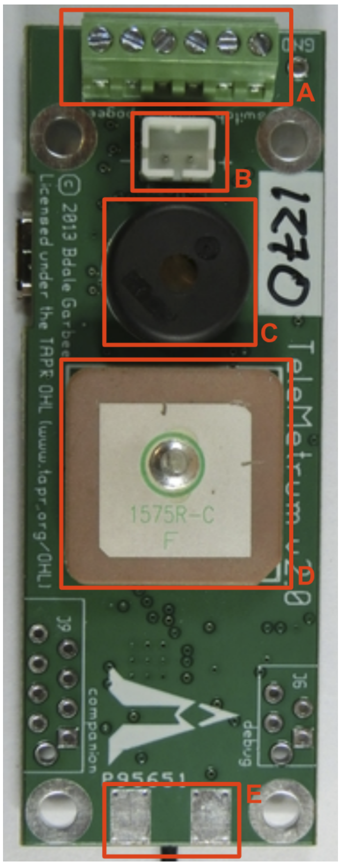


Fig. 11: The flight computer used in the preliminary and final designs. Components highlighted: A. Switch and charge outputs on a terminal B. JST battery connection pins C. Piezoelectric buzzer for communicating board status D. GPS antenna E. Radio transmitter antenna

More information about the flight computer's performance in the rocket is now known. Figure 11 shows the terminal output and inputs (A) to which wiring will be attached for the main power switch and the charge canisters. The power switch will be mounted externally on the rocket, as per the rules of the competition, and will have 2 leads connected to two of the terminals ports in the figure. The

team must be able to turn the flight computer on or off once the rocket is fully assembled and on the launchpad.

Choosing a switch for the rocket may seem rather trivial, however it too has performance requirements to consider, as specified in section III-D. In order to prevent the it from being toggled by external forces, a rotary switch was chosen over a simple slide or lever switch, as pictured in Figure 12.

Fig. 12: The rotary switch chosen for turning the Telemetrum on/off.

The rotary switch has a mechanism which provides resistance against movement between positions, so it has greater pole and throw capabilities than a simpler slide switch might have. The rotary switch will always be physically pulled towards a position whereas a slide switch may be placed between two positions. Lever switches also contain a similar mechanism, however they can easily be triggered by external forces. These forces may be capable of toggling the lever switch in a linear motion, but they would not be able to produce a torque large enough to actuate a rotary switch. Thus the rotary switch is the most robust option for mounting externally on the rocket.

C. *Flight Computer Implementation*

The flight computer is mounted within the rocket according to a design used by last year's Union College Rocket Team [7]. The Telemetrum and battery is mounted on a piece of plywood with runners on either end. Two steel rods mounted between two bulkheads in a compartment of the rocket body pass through runners, allowing variable positioning of the flight computer within the rocket. Figure 13 deconstructs this mounting system so all the parts are visible.

The pressure within the flight computer's compartment must remain equalized with atmospheric pressure to obtain accurate altitude measurements. Other compartments in the rocket must be pressurized however, as the charges used to separate the rocket for parachute deployment will use the pressurization to shear the body. The bulkheads on either end of the runner holding the flight computer serve the purpose of keeping the other compartments from being depressurized due to the equalized pressure in the electronics compartment. Wires will be run from this compartment to the all compartments of the rocket which contain a charge for separating the body.

Fig. 13: The complete avionics compartment in the rocket, deconstructed.

D. Scientific Payload Final Design

While implementing the in air deployment system for HARVe on the rocket, concerns about the chaotic nature of the deployment were brought up among the team. The details of this system are not included in this report, as it is not ultimately a sole responsibility of this project. Due to these concerns though, it was requested that the functionality of HARVe be extended to include a ground deployment as well. The sections to follow exist to explain how the design of the rover is impacted by this request, as well as other changes that were made to HARVe since the preliminary design was written. The original in air deployment design is included in this report even though the implementation of the project may not follow this design. This was done because it was one of the original design requirements of the system, and to preserve the research for future teams which may wish to revisit in air deployment.

1) HARVe - Electronics Design: The behavior of the sensors on board HARVe is most effected by the switch to ground deployment. Instead of imaging the ground during descent from altitude, the camera will now be used to take pictures of the landscape from the ground. The camera will still not be turned on until the photoresistor detects suitable light for the image, as in the preliminary design, but the BME680 sensor will record air quality and atmospheric conditions starting from 1000 ft altitude. The early readings will serve as a calibration period for the sensors, so that when it is deployed on the ground its readings will be reasonably accurate. The ground deployment does not change the preliminary design much, however other design changes are made in the final design for practical reasons.

To conserve space within HARVe's body, the Arduino Uno used in the preliminary design has been changed to an Arduino Nano. This microcontroller retains all the same ports as the Uno, but is significantly smaller. Despite this, the addition of a GPS receiver to the schematic in Appendix A for recovery of the rover requires more ports than either the Uno or Nano has. The schematic has been updated In Appendix B to include the Arduino Nano microcontroller, the DIYmall 6M GPS module, and a shift register to extend the output ports on the Nano to accommodate the additional sensor.

In an effort to organize the increasingly complex circuit diagram in Appendix B, the electronics on board HARVe have been broken into 3 subsystems which handle particular functions of the rover. All subsystems will be assembled and run on the same microcontroller, however the schematics of these subsystems in Appendices C, D, and E respectively each have their own microcontroller for readability.

Motors Subsystem:

The motors subsystem, as shown in Appendix C, contains only the MC33926 Dual Motor Driver, Pololu 131:1 DC geared motors, and 12V battery. The functionality of these components is no different than described in the preliminary design.

Sensors Subsystem:

Appendix D depicts the sensors subsystem, which simply contains the Arducam camera module, MicroSD card reader/writer, and Adafruit BME680 sensor. As mentioned, the BME680 will begin taking readings at 1000ft to calibrate the sensors. From launch, the barometric altimeter on the BME680 will take altitude measurements at a very low sampling rate, such as 0.25Hz. HARVe will then be able to detect when it reaches 1000ft during descent, and at this point, the microcontroller will ”wake” the other sensors, begin recording and transmitting the data, and increase the sampling rate. The camera will only start taking photos after deployment on the ground. Ground state will be determined by a combination of the altimeter, accelerometer, and photoresistor. This subsystem is one of the most affected by the switch from an in-air deployment to a ground deployment.

Communications Subsystem

The last subsystem can be seen in detail in Appendix E, and contains the components used to communicate with external systems, as well as a few miscellaneous components. The notable changes in the final design are in the addition of the 74HC595 Shift Register and the GPS module. The shift register buffers inputs from 3 digital output pins on the Arduino, and extends them to 7 of its own independently controlled digital outputs. The register is used to send data to the HX1 radio transmitter and DIYmall GPS module. The components in this subsystem are largely unchanged in function from the preliminary design.

2) HARVe - Structural Design: The structural design of HARVe has not changed much as a result of the switch to ground deployment, but a few changes have been made to the body as a result of early testing of assembly and movement. The final structural design of the rover body can be seen in the CAD drawing in Figure 14.

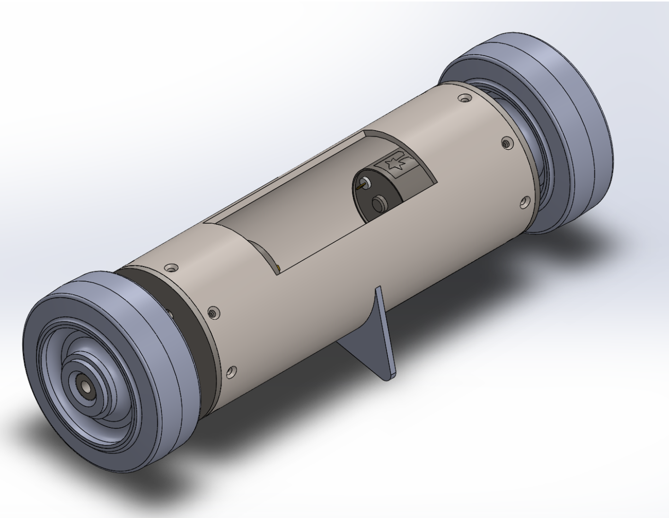


Fig. 14: The final CAD model of HARVe's structural design.

The noteworthy changes to the preliminary structural design detailed in section V-B1 are the replacement of the deployable skid with a rigid tail-dragger for stability, and a slight redesign of the caps which sit in either end of the steel body tube, and in which the DC motors are mounted.

The end-caps have been redesigned to include a lip which will allow them to lie neatly on top of the body tube, instead of having the outer face of the end-caps resting flush with the end of the body tube. This makes the radial screw holes on the body tube and end-cap easier to line up. This redesign was incorporated into the final design when assembly tests of the rover structure revealed it to be an aid to the overall complexity of building it.

The tail-dragger was added to the design during testing of HARVe's movement, during which significant sway was produced in the rover body when the motors would abruptly start and stop turning. To eliminate the sway, the triangular tail-dragger, as seen in Figure 14, protrudes from the rover body and will stabilize movement of the rover as it drives by dragging on the ground. The protrusion is still within size constraint of the 3U CubeSat.

Figure 15 is an image of HARVe with the preliminary and final design structural elements fully implemented, and the sensors and motors subsystems operational on-board. This is a prototype of the final rover, so the microcontroller and breakout boards are installed on breadboards, making them appear quite a bit larger than they will be when soldered together in the final construction.

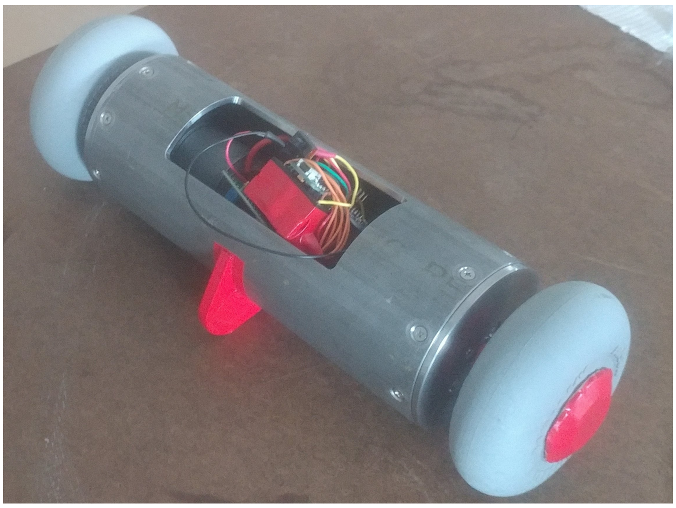


Fig. 15: Construction of HARVe as detailed by the preliminary and final designs.

Another stability measure being considered, however not yet fully integrated into the design, is the extension of the wheel treads to cover the majority of the rover body, as demonstrated in Figure 16.

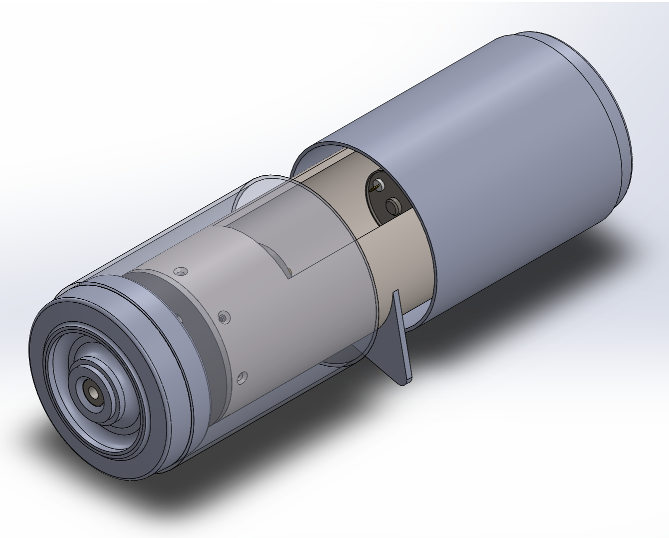


Fig. 16: The final CAD model of HARVe's structural design with treads extended from the wheels towards the center of rover body. Note transparency on near tread.

The rigid extended treads improve the structure of HARVe in 2 ways. First, it adds a layer of protection to the electronics and weaker structural points, such as the motor shafts. Next, it means that a much larger powered surface will be in contact with the ground at any given time, improving its ability to surmount obstacles in the terrain. These are both advantages of the structural addition regardless of whether it is deployed in-air or on the ground, however testing remains to done on HARVe's ability to surmount obstacles without the treads before implementing them in the final design.

VII. Preliminary Testing Results

The Rocket Team constructed a small model rocket from a kit and designed an adapter such that the Telemetrum could fit within the model rocket without interfering with its functionality. The purpose of the test was to launch the model and evaluate the performance of the flight computer. A successful test would mean that all operational functions of the Telemetrum work as expected, including the launch trigger, altitude and all derived measurements, and GPS location. Radio transmission was not tested. After a successful launch of the model with the flight computer on board, the Telemetrum was recovered and the data was downloaded for analysis. The data collected by the Telemetrum includes, but is not limited to the data graphed in Figure 17.

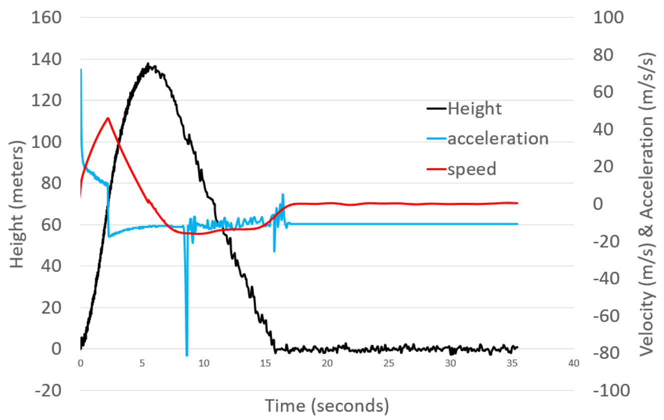


Fig. 17: A plot of the height, acceleration, and speed of a model rocket during a test flight.

The test revealed that the flight computer recorded all data as expected, and verified that the computer could be trusted. Note the sudden dip in acceleration, this is likely due to the fact that the two

halves of the rocket body unexpectedly separated as they came down to Earth, and is not a sensor error. The Telemetrum was triggered by the massive spike in acceleration at the beginning of the flight, and from then on all data was recorded as expected. Figure 18 is displays the path taken by the rocket as seen by the GPS satellite. Takeoff and boost phase occurred where the path is colored red, coast phase occurred where the path is colored yellow, descent followed the blue path, and finally the rocket came to rest on the black marker.

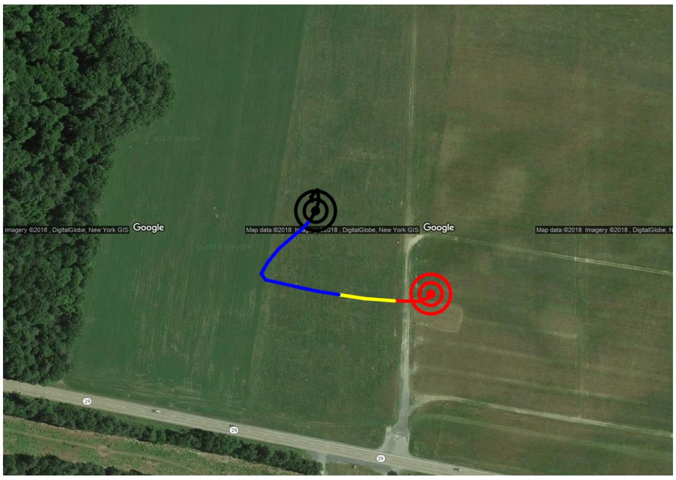


Fig. 18: GPS tracking of model rocket test flight

The Telemetrum has also been demonstrated to be capable of separating the rocket at decoupling points. Multiple ground test have been performed in which the detonation of a black powder charge was successfully remotely detonated to shear the pins connecting two compartments of the rocket body. Using the results from these tests and the ideal gas law to calculate the theoretical amount of black powder needed to shear the pins, the team is able to better understand how calculated values compare to real world data.

Based on the limited testing performed thus far, the Telemetrum flight computer has performed extremely well. More launch tests will need to be planned to experiment with range and telemetry, but the ease of use and accuracy of the sensors has been made apparent in early testing.

VIII. Performance Estimates and Results

A. Flight Computer Performance

The Telemetrum flight computer has thus far performed all of its functions accurately and fulfilled all design requirements. Preliminary testing data has confirmed the accuracy of the sensors on board. The design requirements of the the flight computer are enumerated below, along with details about the Telemetrum relevant to the requirement.

1) COTS System - The Telemtrum is an OTS component
2) Radio and GPS Communication - The Telemetrum has a GPS antenna for simultaneous communication between up to 3 satellites, and a 433.3 MHz radio transceiver with a theoretical range of 100,000 ft.
3) Telemetry - In addition to storing the flight information on-board, the Telemetrum will transmit flight data and recovery information live to a ground station.
4) Dual Deploy- The Telemetrum has 2 sets of ports for connecting leads to a black pow-

der charge. This enables 2 independently deployed parachutes to be timed over the course of the flight.

5) Power - The single cell, 900 mAh battery recommended for use with the Telemetrum can power the flight computer continuously for 6 hours at peak load.
6) Power Switching - The Telemetrum contains a set of ports allowing for connection to an externally mounted master switch on the rocket. This allows it to be turned on and off while on the launch pad, sealed within a fully assembled rocket.

The Telemetrum is a flight computer which is well suited to the mission specified by the IREC Competition Rules. In experiments done since the first round of preliminary testing, data recovery via radio transmission and use of the flight computer for rocket body separations has been confirmed to function as expected, however these tests were performed on the ground, and a full scale launch has yet to be performed. Based on these ground tests though, it is estimated that the flight computer will be quite reliable when launched. In fact, it is recommended that the Telemetrum V2.0b be used by any future teams competing in the SA Cup Competition under similar requirements.

B. Scientific Payload Performance

The desired functionality of HARVe was specified in Table III. In Table V, these requirements are repeated, along with the status of their implementation.

Function	**Implementation**
Detect deployment from rocket	Partial
Photograph terrain during descent	None
Detect landing	Partial
~~Deploy skid and decouple parachute~~ Stabilize rover movement	Full
Explore landing zone in a spiral	Partial
Evaluate temperature, humidity, Pressure, and VOC Gas content	Full
Transmit data to ground station	None
Log data	Full

TABLE V: Evaluation of the current prototype of HARVe by the earlier specified functional decomposition.

In the paragraphs to follow, HARVe's ability to perform each of these functions as currently prototyped and as planned will be described.

HARVe is presently only partially capable of detecting deployment from the rocket, as not all of the sensors used to detect deployment have be tested. The altimeter functions in the current prototype, however the photoresistor and accelerometer have not yet been programmed in. These components are completely worked into the full design though, and can be built into the next prototype of the HARVe.

Referring two the second item in the table, there is currently no implementation of the camera module in the prototype. There is a plan to implement such a sensor though, and the desired sensor has

been purchased and is included in the electrical schematic. The camera will begin taking pictures once the the photoresistor detects a suitable amount of light after deployment, and will send its images to be stored on the MicroSD card, not to be transmitted via radio.

The third item in Table V, detect landing, is now tied up with detecting deployment due to the transition from an in-air to ground deployment system. Since the detect landing functionality will ultimately be the trigger for the deployment mechanism in the ground ejection system, its implementation in the current prototype is at the same stage, partial completion.

The fourth item in the table has been altered to reflect the final design changes. The skid is no longer electronically deployed, but is now a rigid protrusion off the rover body. Additionally, since HARVe is deployed on the ground, there is no need for a mechanism to decouple a parachute. Thus, the function has been replaced by the the usage of any stability enhancing structural components on the rover. Since the ”tail-dragger” has been completely built and tested in the current prototype, it is considered fully implemented.

Referring now to the fifth item in Table V, the current rover is capable of driving itself in a straight line, but not an exploratory spiral. Thus, it is partially implemented, as HARVe is capable of performing the task with its current hardware, however the particular behavior has not been programmed.

The sixth item in the table, the habitability assessing functions, are considered fully implemented. The hardware is properly installed in the current prototype, and the rover is programmed to read in, analyze and store data from the array of sensors.

Radio transmission of data is in a similar state to the camera module, in that it has no implementation in the current prototype, however the hardware has been purchased and is accounted for in the design of the next prototype.

The final item in Table V refers to the rover’s ability to log data on-board, which is entirely working in the current prototype, and is thus fully implemented. Figure 19 is an image of the earlier prototype of the data-logging and habitability evaluation circuit. The code for taking a reading of the quantities mentioned in Tables III and V and writing it to a MicroSD card is included in Appendix F.

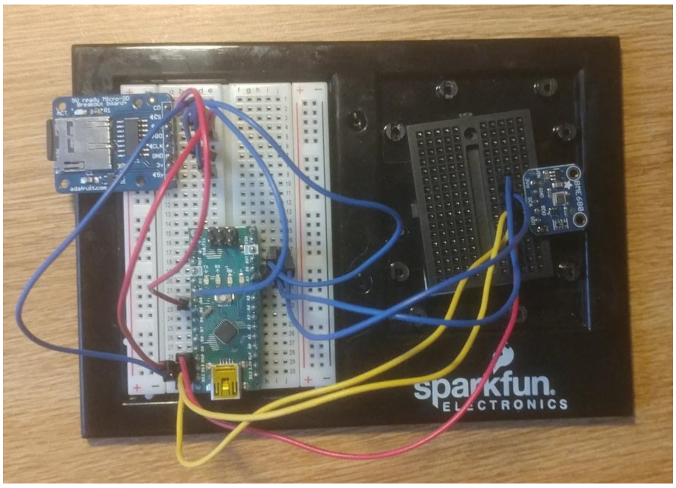


Fig. 19: Prototype of the sensors subsystem implemented in HARVe.

IX. Production Schedule

The competition for which the project was designed does not occur for another 3 months at the time of submission of this report, and for this reason it was always intended that the project continue beyond what would be covered by this paper. At the time of submission, the flight computer has been installed in the rocket and thoroughly tested on the ground, and HARVe exists as a prototype capable of motor functions, data logging, and habitability assessment. Functionality of HARVe will be continued and completed in the months following the submission of this report and preceding competition in June of 2019.

Conception and design of this project began in April of 2018. Originally, the project only detailed the design of a flight computer for the Union College Rocket Team. This design and proposal was solely developed over the course of several months, until October of 2018, at which point the decision was made to design a scientific payload for the Union College Rocket Team in addition to the flight computer. The idea to make the scientific payload a rover was made in the same month, and so HARVe was fully conceptualized and detailed design began by November 2018. From these designs, the first edition of this report was written, and the implementation schedule as depicted in Appendix G was drafted.

The months of January 2019 to March 2019 were devoted to sourcing components to build both parts of the project and to construction of prototypes. The submission date of this report is March 21st, 2019. Future work on the project for the most part solely remains in the completion of HARVe. Construction has begun on the second prototype of the rover, which hopes to fully implement the remaining functionality detailed in Table V.

X. Cost Analysis

A complete breakdown of the components used for the flight computer system, HARVe, and the project as a whole is included in Appendix H. Approximately 89% of the cost of the flight computer system is made up by the Telemetrum itself and the TeleDongle, a USB peripheral which is configured to receive the radio messages transmitted by the Telemetrum. Since the project is not a product for sale, the cost of the flight computer system is a reflection of the market price of the sum of its components. The same is true for the cost of HARVe. The cost of HARVe is split relatively evenly among its components, with no single item making up a majority of the cost.

In total, the project is estimated to have cost $892.92, with a 56/43 split between the flight computer system and rover costs, respectively. The total cost is quite reasonable for undergraduate engineering competition team, and is covered in large part by a Student Research Grant given to the Union College Rocket Team, and with funds granted by

the Union College Mechanical Engineering Department.

XI. Conclusion

A. Problem Summary

The problem as originally described is derived from the Union College Rocket Team's international competition, and has two components to it:

1) The Union College Rocket Team needs a device to log and transmit flight data on-board a model rocket in accordance with competition rules.
2) The Union College Rocket Team needs a payload of scientific merit to carry on board a model rocket.

The first problem, being constrained by competition rules, is solved for the most part by reliance on a single design decision, that being what COTS flight computer to buy. If a computer is chosen which conforms to the IREC rules and works out-of-the-box, not much can be done to improve its functionality beyond that. The design process to solve the scientific payload problem has not such constraints though, so all performance goals are imagined and set by the team. This means there is a lot more design involved in solving the problem, leading to a very interesting project overall.

B. Designs Offered for Problem Resolution

The design performance of the flight computer specifies its main duties as logging rocket altitude on board for competition scoring, Transmitting GPS location via radio for remote recovery, and deployment of parachutes in the rocket in two stages for reduced drift. The Telemetrum V2.0b flight computer was chosen to solve this design problem, as its built in components were capable of accomplishing all of these tasks. The Telemetrum has proven to be a robust and effective choice in low altitude flights and ground tests, and will be tested in a full scale launch of the competition rocket in May of 2019. In future years, it will perhaps be worth it to invest in more advanced flight computers such as the TeleMega, which is capable of deploying up to 6 parachutes individually, as opposed to the Telemetrum's 2. The extra terminals do not have to be used for parachutes, though, as having additional outputs on the flight computer opens up many new possibilities for additional valuable data collection systems to be installed on the rocket.

To solve the problem of supply the team with a scientific payload, the Habitability Assessing Research Vehicle (HARVe) was hypothesized, designed, and prototyped. The scientific achievement of HARVe is the assessment of the habitability of the landing zone of the rocket, primarily through imaging and air quality analysis. Initially, HARVe was designed to be deployed during the flight of the model rocket, at apogee. Concerns about the chaotic nature of such a deployment potentially harming the project as a whole encouraged a design be pursued

where the rover would be deployed on the ground after rocket touchdown. The design of HARVe is flexible enough that this did not necessitate many design changes to the original in-air design, and in fact, the rover could easily fulfill either function should either the in-air deployment problem be solved, or the ground deployment solution carried forward.

C. Closing Remarks

Splitting the project into two components, the design of the rocket's on board flight computer and the design of the scientific payload, brought an interesting perspective. On one hand, it seemed that the design of the flight computer progressed much quicker than the development of HARVe. Being constrained to a very explicit set of rules meant that, ultimately, not as much thought was necessary to carry out a design which fit those rules exactly. HARVe, on the other hand, progressed much slower, and was perhaps hindered by unrealistic hopes for the design early on. It is important, when there is as much freedom in a design problem as was allowed in the development of HARVe, that the requirements be stated as explicitly and thoroughly detailed as possible. Thinking about such things early on may have provided insight into what the real problems were in developing the prototype, and more time could be allocated to such problems instead of extraneous features. The project was an excellent learning experience in that sense, and working with a team certainly helped with design process substantially.

References

[1] Project Apollo: A Retrospective Analysis https://history.nasa.gov/Apollomon/Apollo.html

[2] Private Space Flight: American Deregulation https://en.wikipedia.org/wiki/Private_spaceflight#American_deregulation

[3] NAR. High Power Rocketry, FAA Launch Authorization https://www.nar.org/high-power-rocketry-info/filing-for-faa-launch-authorization/

[4] ESRA. Spaceport America Cup Official Rules 2019 http://www.soundingrocket.org/uploads/9/0/6/4/9064598/sa_cup_irec_rules___requirements_document_20181103__rev_b_.pdf

[5] ISM Bands: Frequency Allocation https://en.wikipedia.org/wiki/ISM_band

[6] HuygensFresnel principle https://en.wikipedia.org/wiki/Huygens%E2%80%93Fresnel_principle

[7] A. Attorri, O. Gauthier, C. Marina, and M. Reinhardt. Design of a Commercial off the Shelf (COTS) Rocket for 10,000 foot Apogee. May 2018.

[8] Las Cruces Weather. Average Maximum Windspeed http://www.lascruces-weather.com/wxwindsummary.php

[9] Electronic Warfare and Radar Systems Engineering Handbook: Power Density http://www.rfcafe.com/references/electrical/ew-radar-handbook/power-density.htm

[10] Texas Instruments. CC1120 High-Performance RF Transceiver http://www.ti.com/lit/ds/symlink/cc1120.pdf

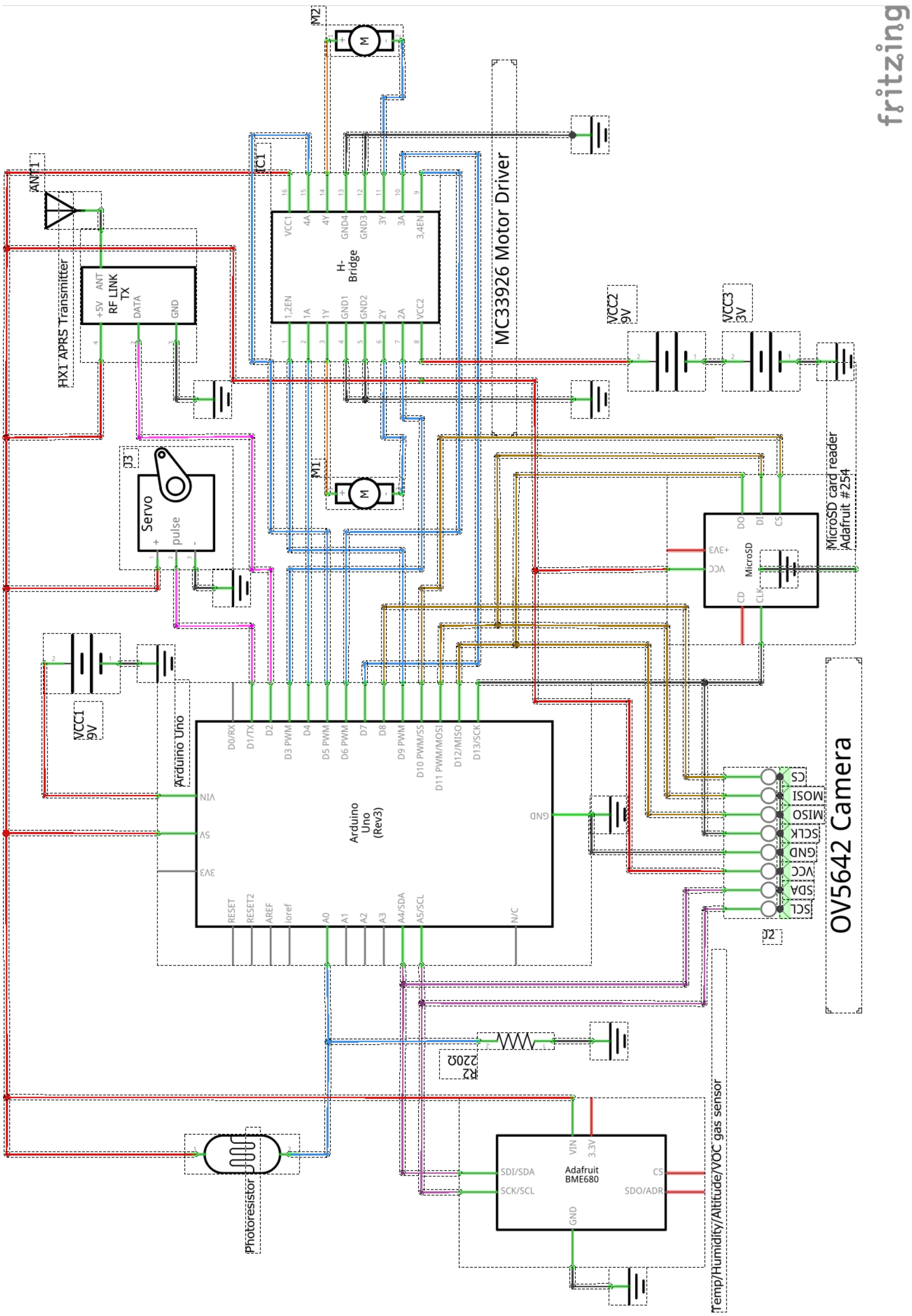
[11] B. Garbee, K. Packard, B. Finch, A. Towns. The Altus Metrum System. September 2016. https://www.apogeerockets.com/downloads/PDFs/AltusMetrumUserManual_20160904.pdf

[12] Ultimate tensile strength: Typical tensile strengths of some materials https://en.wikipedia.org/wiki/Ultimate_tensile_strength

[13] HX1 VHF Narrow Band FM 300mW Transmitter
https://cdn.sparkfun.com/assets/d/1/1/4/b/HX1_APRS.pdf

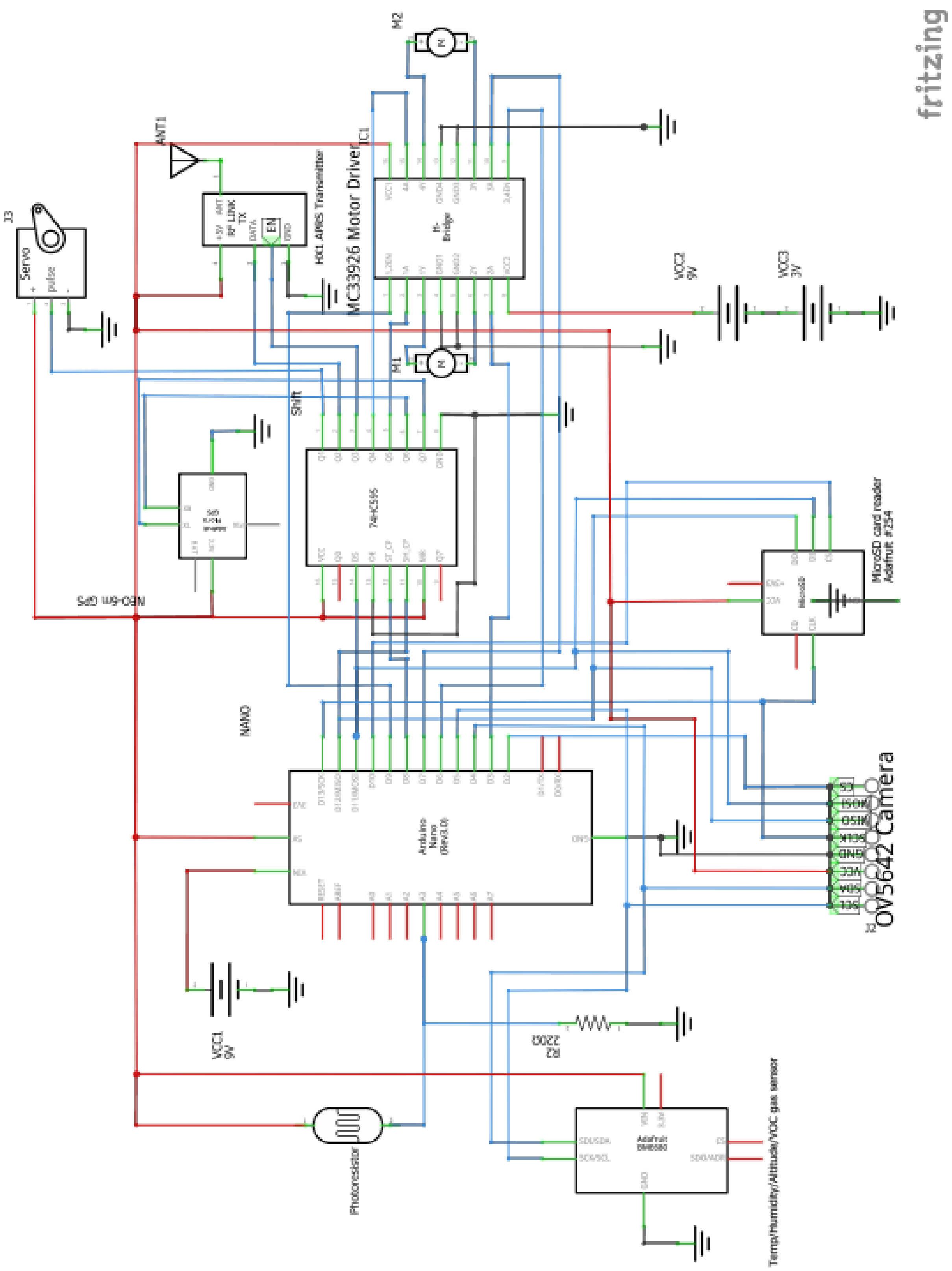
APPENDIX A

HARVE ELECTRONICS PRELIMINARY DESIGN FULL SCHEMATIC



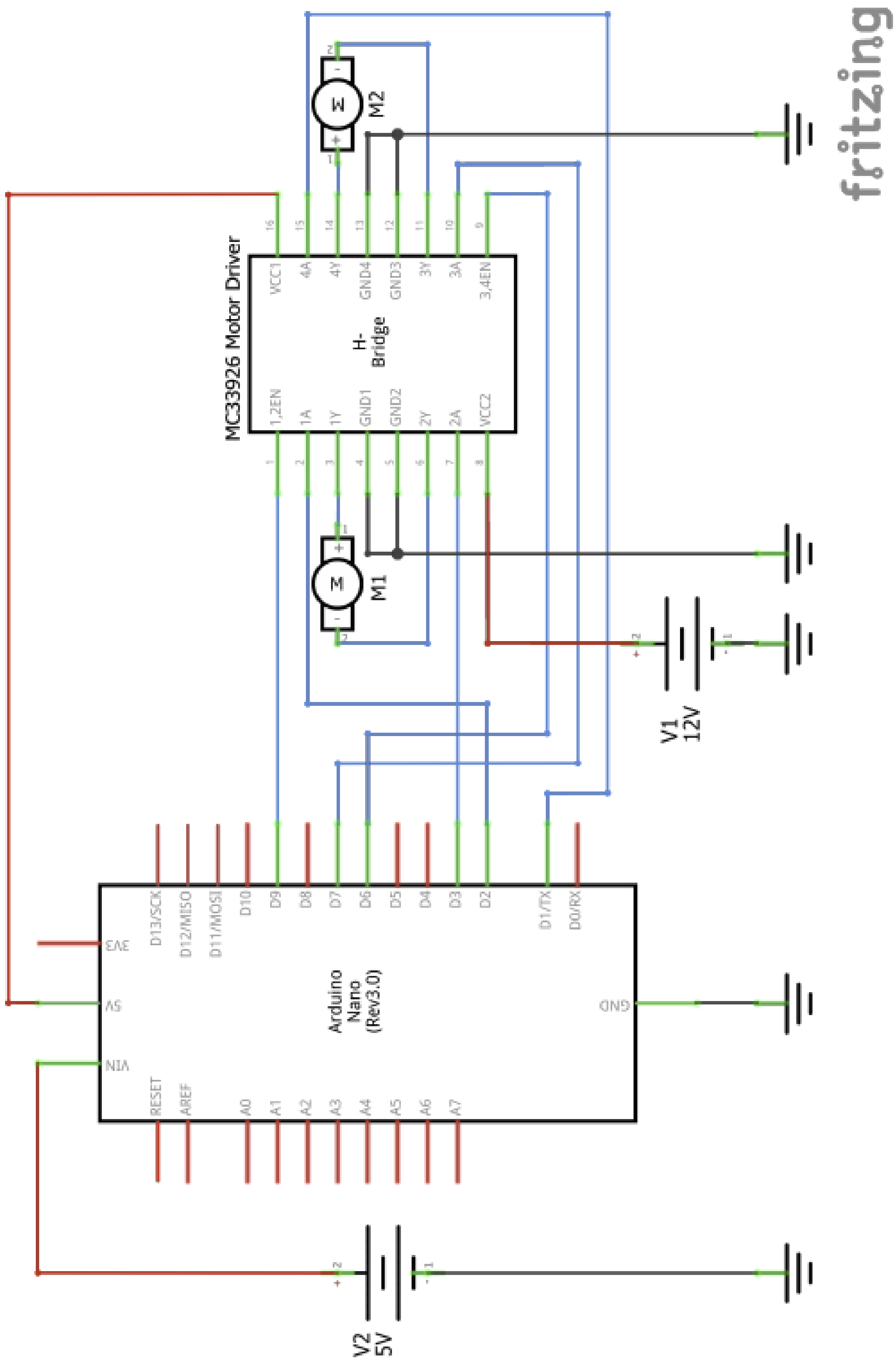
APPENDIX B

HARVE ELECTRONICS FINAL DESIGN FULL SCHEMATIC



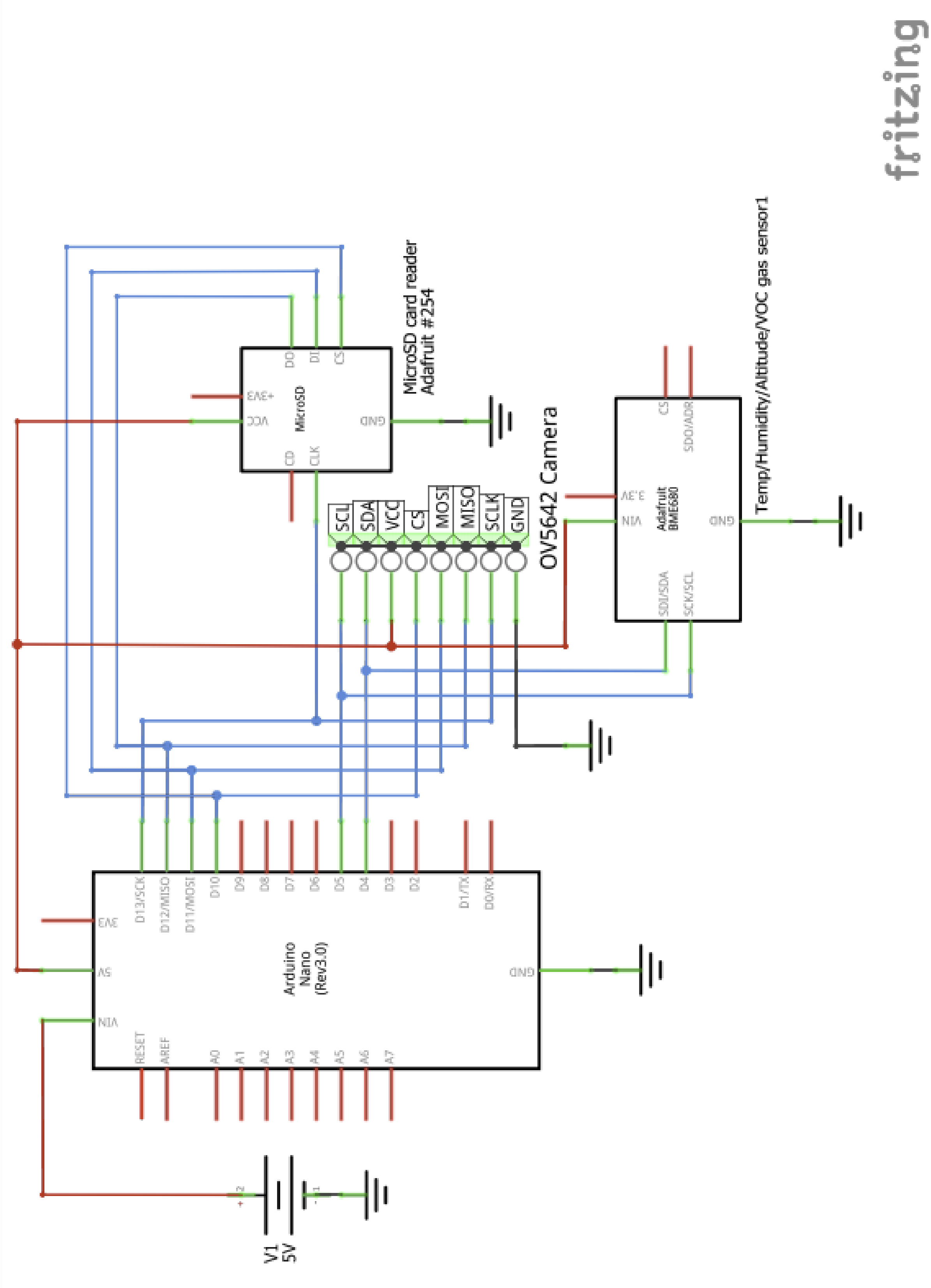
APPENDIX C

HARVE ELECTRONICS MOTOR SUBSYSTEM SCHEMATIC



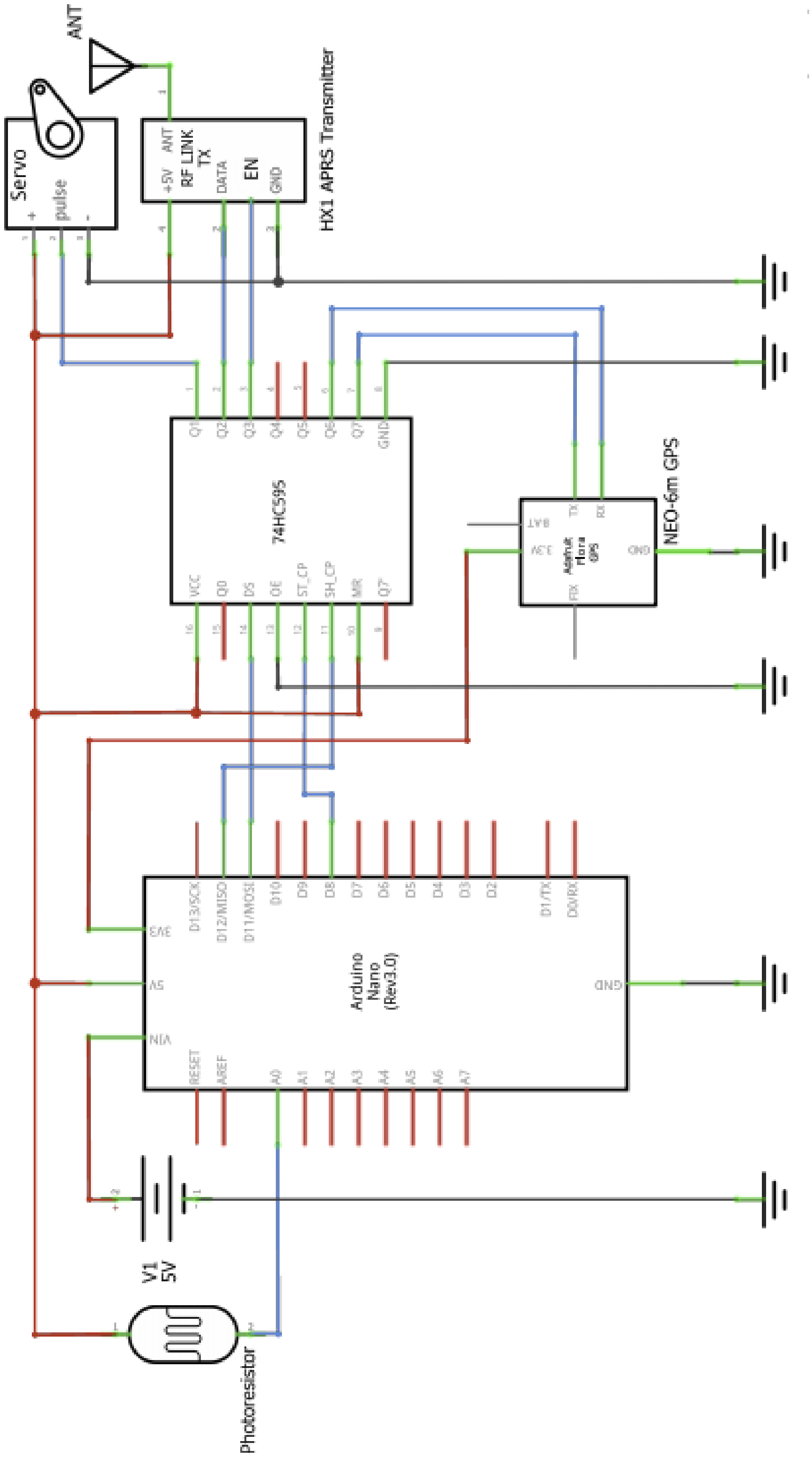
Appendix D

HARVe Electronics Sensors Subsystem Schematic



APPENDIX E

HARVE ELECTRONICS COMMUNICATIONS SUBSYSTEM SCHEMATIC



APPENDIX F

ARDUINO CODE: READING IN HABITABILITY VARIABLES, STORING ON MICROSD CARD

```
#include <Wire.h>
#include <SPI.h>
#include <SD.h>
#include <Adafruit_Sensor.h>
#include "Adafruit_BME680.h"

Adafruit_BME680 bme(10, 11, 12, 13);

void setup() {
  // Open serial communications and wait for port to open:
  Serial.begin(9600);
  while (!Serial);

  Serial.print(F("Initializing BME680..."));

  if (!bme.begin()) {
    Serial.println(F("Could not find a valid BME680 sensor, check
       ↪ wiring!"));
    while (1);
  }
  Serial.println(F("BME680 initialized"));
  pinMode(10, OUTPUT);
  digitalWrite(10, HIGH);

  Serial.print(F("Initializing SD card..."));

  // see if the card is present and can be initialized:
  if (!SD.begin(4)) {
```

```
    Serial.println(F("Card failed, or not present"));
    while (1);
  }
  Serial.println(F("card initialized."));

  // Set up oversampling and filter initialization
  bme.setTemperatureOversampling(BME680_OS_8X);
  bme.setHumidityOversampling(BME680_OS_2X);
  bme.setPressureOversampling(BME680_OS_4X);
  bme.setIIRFilterSize(BME680_FILTER_SIZE_3);
  bme.setGasHeater(320, 150); // 320*C for 150 ms
}



void loop() {
  Serial.println(F("done"));
  // make a string for assembling the data to log:
  digitalWrite(10, LOW);
  String dataString = "";
  if (!bme.performReading()) {
    Serial.println(F("Failed to perform reading :("));
    return;
  }
  Serial.println(F("done"));
  dataString += bme.temperature;
  dataString += ", ";

  dataString += bme.pressure/100.0;
  dataString += ", ";
```

```
  dataString += bme.humidity;
  dataString += ", ";

  dataString += bme.gas_resistance/1000.0;
  dataString += ", ";

  dataString += bme.readAltitude(1013.25);

  // open the file. note that only one file can be open at a time,
  // so you have to close this one before opening another.
  File dataFile = SD.open("test.txt", FILE_WRITE);
  Serial.println(F("done"));
  // if the file is available, write to it:
  if (dataFile) {
    dataFile.println(dataString);
    dataFile.close();
  }
  // if the file isn't open, pop up an error:
  else {
    Serial.println(F("error opening datalog.txt"));
  }
  Serial.println(F("done"));
  delay(2000);
}
```

APPENDIX G

IMPLEMENTATION SCHEDULE

	Winter Break	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10
Electronics and Payload											
- Flight Computer											
- Range testing										Testing of altimeter capabilities not high priority Winter term	
- GPS location testing											
- HAM Radio License	Obtain license over break in the event that range testing can be done sooner										
- Payload											
- Electronics Schematic	Prepare to order ASAP Winter term										
- Electronics Ordering		Confer with rocket team									
- Testing and Materials analysis			Prepare testing	Motor Testing		Sensor Testing		Test motors installed on rover		Install all electronics in rover and debug	
- Structures Ordering				Confer with rocket team							
- Manufacture						Build rover structure	Install motors on rover structure	Assemble complete electronics package	Program independent functions into a single program		
KEY:	**Planning/Prep**	**Testing**	**Manufacturing**								

APPENDIX H

PROJECT COMPONENT COSTS

Flight Computer					
#	Part	Retailer	Price Per	Quantity	Total Price
1	Telemetrum V2.0b	Apogee Components	$346.15	1	$346.15
2	TeleDongle V3.0	Apogee Components	$100.00	1	$100.00
3	900 mAh LiPo	Apogee Components	$11.54	1	$11.54
4	XL Ejection Canister	Apogee Components	$3.30	3	$9.90
5	Wood/PVC	HobbyTown	$15.00	1	$15.00
6	Wiring/Hardware	McMasterCarr	$20.00	1	$20.00
TOTAL					**$502.59**

HARVe					
#	Part	Retailer	Price Per	Quantity	Total Price
1	Photoresistor	Adafruit	$0.95	1	$0.95
2	Shift Register	Adafruit	$2.75	1	$2.75
3	Temp/Humidity Sensor/Pressure/VOC Gas	Adafruit	$22.50	1	$22.50
4	SD card reader	Adafruit	$7.50	1	$7.50
5	Arducam 5MP Sensor	Amazon	$39.99	1	$39.99
6	GPS Module	Amazon	$15.99	1	$15.99
7	Arduino Nano	Arduino	$22.00	1	$22.00
8	Wiring/Hardware	McMasterCarr	$20.00	1	$20.00
9	Steel tube	OnlineMetals	$59.26	1	$59.26
10	Steel Plate	OnlineMetals	$13.12	1	$13.12
11	Steel Round Bar	OnlineMetals	$3.55	1	$3.55
12	DC Motor Driver	Pololu	$29.95	1	$29.95
13	Colson Performa 3.5" Wheels	The Robot Marketplace	$5.99	2	$11.98
14	DC Motor	Robotshop	$24.95	2	$49.90
15	HX1 Narrow Band Transmitter	Sparkfun	$50.95	1	$50.95
16	Additional Stock Material	N/A	$40.00	1	$40.00
TOTAL					**$390.39**

Flight Computer					
#	Part	Retailer	Price Per	Quantity	Total Price
1	Flight Computer	Various	$502.59	1	$502.59
2	HARVe	Various	$390.39	1	$390.39
PROJECT TOTAL					**$892.98**