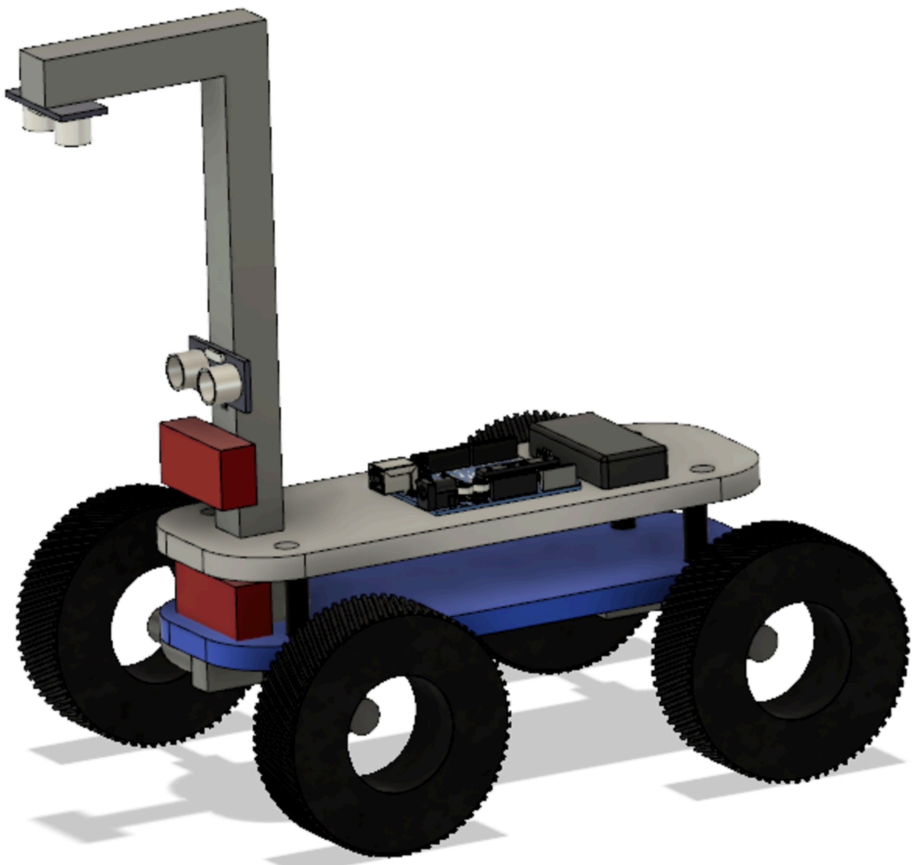
Crash Site Milestone 3

Team Johnny Crash

ENES100 Section 0502

15 October 2021



[1]full CAD drawing

Table of Contents

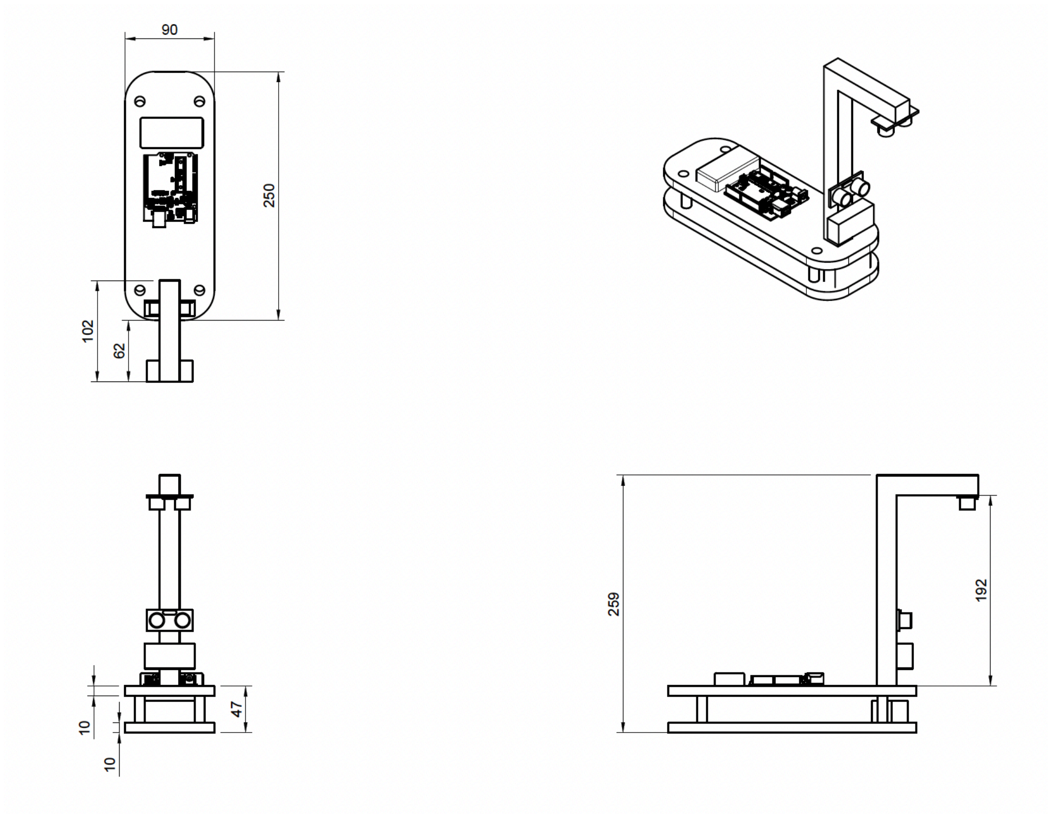
Executive Summary

essential descriptions for someone to (i) understand the context and the most basic constraints of the design problem, (ii) imagine the composition and layout of your OTV, and (iii) understand how your OTV will carry out the mission. Do not simply repeat the mission objectives in the product specification document.

We have been tasked to design and build an autonomous vehicle that can complete a mission and navigate its way through a course. The mission we are required to complete is to survey the crash site, measuring the dimensions of a rectangular prism and identifying if there is an abnormality, and if so, on which side of the prism. To complete the mission, our vehicle will use a combination of ultrasonic and infrared sensors. The ultrasonic sensors will be used to measure the dimensions of the wreckage. The infrared sensors will be used to detect the side of the wreckage with an abnormality.

Chassis

The chassis of the vehicle faces two main challenges - one, that the chassis must be long enough to realistically hold all of the components necessary for the OTV's operation, yet must also be small enough to avoid getting stuck while crossing the log or other two obstacles. Additionally, the chassis must have a total width such that it may cross through small or tight corridors that may be present on the field. As for our team requirements, we are expected to maintain a vehicle footprint no larger than 300x300mm, and a mass not exceeding 2.5 kg. Our navigation team requested our total OTV width to be <250mm in order to pass between the obstacles and the arena wall. Additionally, we figure we need our wheels (49mm radius,) to be at least 3.5 cm from our chassis to allow for sharp turns. This leaves our chassis itself to be 90mm wide. Our solution to both the challenges and the requirements is the use of laser-cut plywood and plexiglass as the bases for a two-layer chassis. In order to assure the laser cut materials are strong enough, we will glue 3 layers of each of the plywood pieces and the acrylic pieces together. Their depths are ~3.2mm, so when glued together, they will each be approximately 9.6mm in depth plus glue will be 10mm. The locomotion and propulsion components will be located in the lower chassis, while the electronics and sensors are located on the top chassis. There will be a long arm to hold the ultrasonic sensor facing down at 19.2 cm above the chassis, plus the wheel radius 4.9 cm, and the chassis height above the wheel 4.7 cm, to total 28.8 cm above the ground (see image 2). The two sensors (infrared and ultrasonic) connected to the same arm below will be attached at ~13cm above the ground. The arm will also be made out of the laser cut acrylic. There will also be a lower IR sensor on the first level of the chassis (see image 1) at approximately 6.5 cm above the ground. The small arm holding it up will also be formed by the laser cut acrylic.



[2] Chassis assembly drawing. Dimensions in millimeters.

Propulsion

In order for the vehicle to complete the mission, it must move around the crash site, and identify which side of the prism is “abnormal.” Another requirement is getting the vehicle to move past the obstacles—whether it is moving around them or over them—including getting over the log. Finally, the last requirement is completing the mission and course in under five minutes.

Some of the challenges of that is making sure that the vehicle is oriented in the correct direction so that the sensors are facing the crash site. We must also find suitable mechanical elements such as wheels, motors, and batteries that will allow the vehicle to fulfill these requirements.

Wheels

1:10 RC Rock Crawler 1.9 Inch Rubber Tires 98MM for Axial SCX10 Tamiya CC01 D90 TF2

- Tire Diameter: 98 mm
- Material: Rubber tire and plastic rim
- Mass: 266.5 g

We plan on having our vehicle going over the log as due to the height of the vehicle, it's unable to go under the limbo. As a result, we need to have relatively large wheels. The log is 2” (50.8 mm) in radius and we therefore chose wheels with a 49 mm radius.

Drive Motors

313:1 Metal Gearmotor 20Dx46L mm 6V CB

- 6 V
- Mass: 47 g
- No-load speed: 45 RPM
- Stall Torque: 127.5 N-cm

The vehicle has two motors, one for the front wheels and one for the back wheels. We have decided to go with two 6V motors as we estimated our minimum torque for steady, flat driving is estimated to be 0.6003 N-cm while the minimum torque to get over the log is estimated to be 41.67 N-cm.

<u>For steady, flat driving:</u>

$m = 2.5$ kg

$r = 4.9$ cm

$C_{RR} = 0.01$

$F_{RR} = (0.01)(24.5N) = 0.245N$

$\tau = (0.245N)(4.9cm) = 1.2005N\text{-}cm$

1.2005N-cm / 2 motors = **0.6003N-cm**

To get over log:

$\theta \approx 60°$

$m \approx 1.0$ kg (over 1 axle)

$r = 4.9$ cm

$C_{RR} = 0.001$

$F_{TF} = mg\cos\theta$

$= (1\text{ kg})(9.81m/s^2)(\sin 60°)$

$= 8.50$ N

$F_{RR} = mg\sin\theta C_{RR}$

$= (1\text{ kg})(9.81m/s^2)(\cos 60°)(0.001)$

$= 0.0049$ N

$F_{TE} = F_{TF} + F_{RR}$

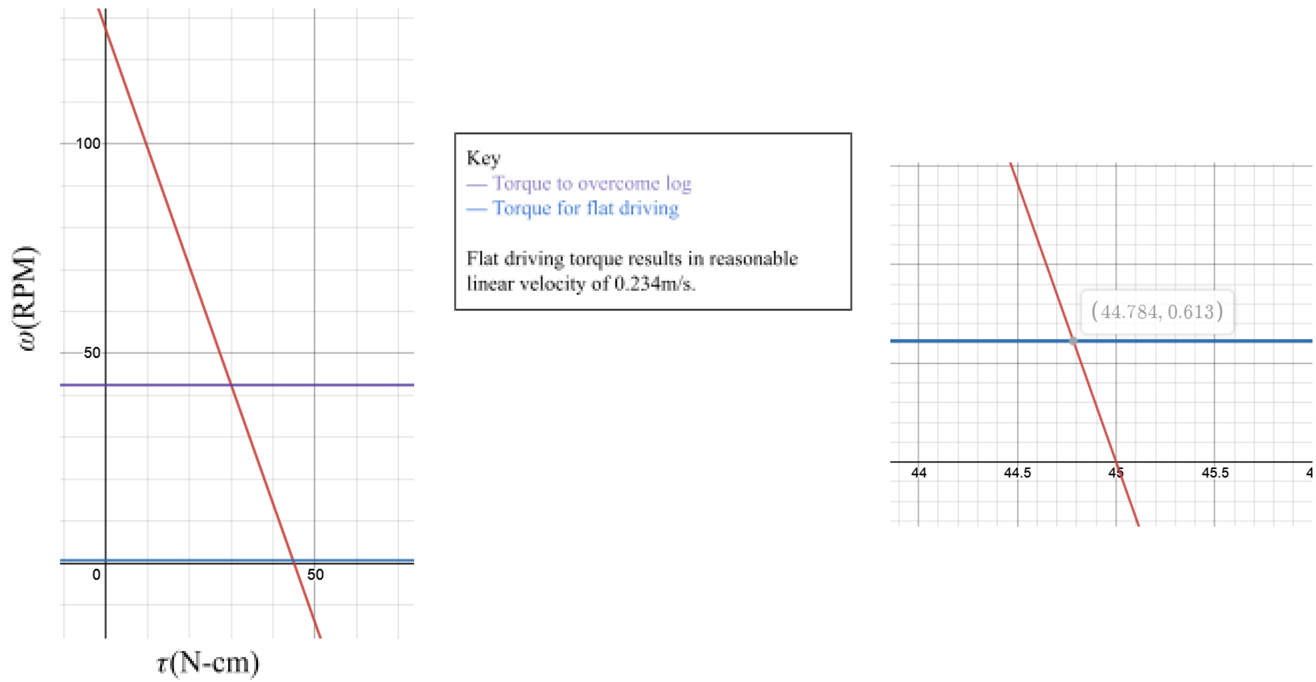
$= 8.50\text{ N} + 0.0049\text{ N} = 8.5049\text{ N}$

$\tau = F_{TE}r$

$= (8.5049N)\ (4.9\text{ cm})$

= **41.67 N-cm**

Motor Characteristics



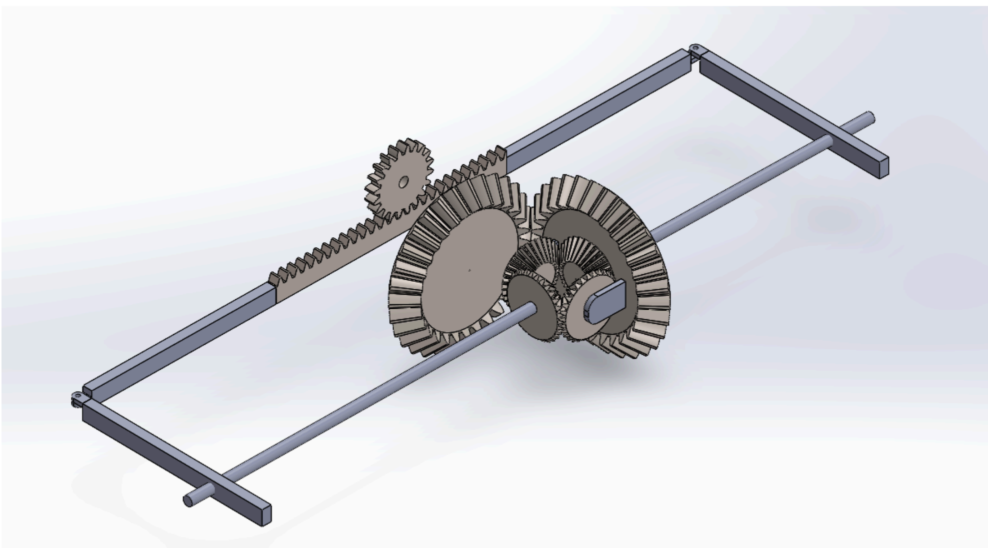
[3] Motor characteristics

Motor Controller: [*Qunqi L298N Motor Drive Controller Board Module Dual H Bridge DC Stepper For Arduino*]

We plan on using two motor controllers, one for the drive wheels and one for the stepper motor. This dual channel H-bridge driver can drive one 2-phase stepper motor, one 4-phase stepper motor or two DC motors. This motor controller has the ability to handle current and can handle different types of motors—specifically both our linear and stepper motors—making it versatile and convenient for us to use.

Steering System

Our vehicle has a rack pinion paired with a differential in the front and rear of the vehicle, where the steering angle is controlled by the stepper motor. We decided to choose this system because it uses less motors, and allows us to have a lot more maneuverability in tight spaces. Having the front and rear wheels turn in opposite directions creates a very tight turning radius that allows us to get around the crash site to measure distance and reflectivity. The system can be compact and fit within the narrow chassis.



[4] Drivetrain CAD Model of OTV

At the rear of the model is the rack and pinion steering system connected to joints on either side. At the front is the differential to allow for even power delivery, which prevents the tires from scrubbing on either side when making turns. In the front axle are 2 constant velocity joints on either side (total of 4), to allow the axle to move properly when turning. On the outside, the wheels are attached. All 3 motors will be attached to the center of the vehicle on the bottom, facing either the front or back, and for the stepper motor, it will be located toward the side of the vehicle.

Stepper Motor

[Nema 14 Bipolar 1.8deg 18Ncm (25.5oz.in) 0.8A 5.4V 35x35x34mm 4 Wires]

This stepper motor is used for steering. There will be a shaft running down the center of the vehicle, connected to both the front and rear rack and pinion systems. The stepper motor runs at 5.4v, 800mA, and has a holding torque of 18N-cm which should be more than adequate to fit our steering needs. It is bipolar which means it spins in both directions, and only uses 4 wires which means it is going to be easier to fit other components onto our arduino.

Power

Our vehicle is required to have a battery that is able to power it for at least 10 minutes. Additionally, the battery must not be lithium or lead acid. One of the main challenges in finding a battery suitable for our vehicle is making sure it has a voltage that can power the vehicle and fit well with the other electrical components so that they don't get damaged.

For our battery, we found that the 6-Cell AA 7.2V Battery Pack with SM-2P Plug 2000mAh Ni-MH High Capacity is best suited for our vehicle. It is rechargeable which is convenient as we will be testing our vehicle many times. It has a voltage of 7.2 V and a capacity of 2000 mAh.

Component	Consumption
Drive Motors (x2)	150 mA (x2) = 300 mA
Stepper Motor	800mA
Arduino	50 mA
Other components	1134 mA
Total:	2284mA

Capacity of battery: 2000mAh

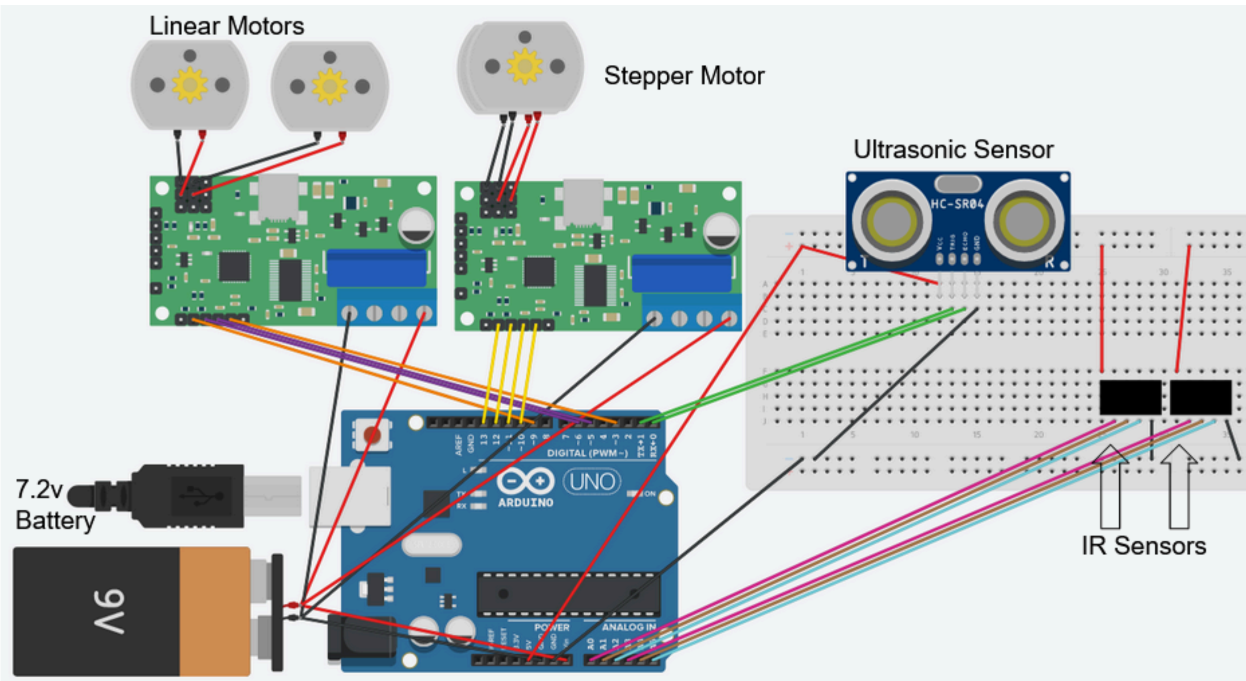
Capacity = (Current)(Time)

2000 mAh = (2284 mA)(T)

T = 0.88 hrs

One concern that was brought up was having a battery too powerful for the motors. Fortunately, the L298N motor controller uses enough power through the H bridge to reduce the voltage to fit within motor specifications. At low amperage, such as the amperage the linear motors will run at, it uses about 1.5v-1.8v, which may affect our top speed somewhat, however, we will still be able to finish the mission on time. At higher amperages, around and above 1A, it uses 2v-3v. The stepper motor is 5.4v and runs at 800mA, so the voltage reduction will be around 2v or more, which should not affect the steering ability much, and ensures that the stepper motor does not break.

Wiring Schematic



[5] Schematic Diagram for OTV

Navigation

The OTV is required to be fully self-operational throughout the entirety of the mission.

This means that we must program our OTV to be able to complete the mission while overcoming the randomized obstacles. To get to the mission zone we are required to avoid two different immovable obstacles, one large barrier and one small barrier with rugged irregular rumbles. To avoid both obstacles we plan on first identifying the south facing wall, and turning the OTV east when we are a specific distance away. The obstacles and the wall can be randomly generated to be a maximum distance of 250mm away from each other, so if we make our OTV slim enough we will be able to drive parallel to the wall to the log and the goal zone, thus avoiding the obstacles all together.

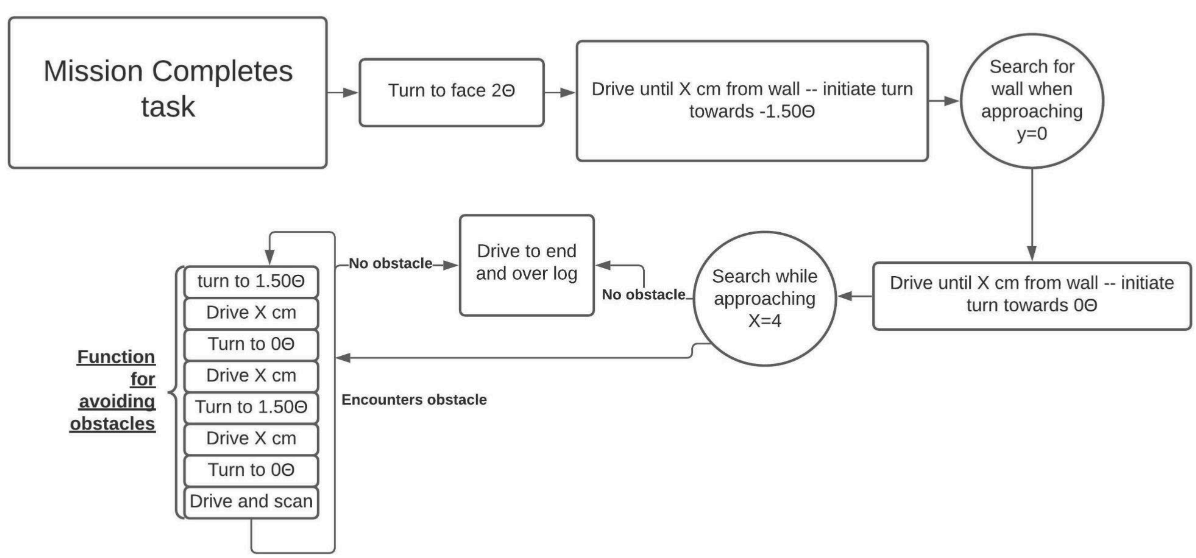
The sensors we will be using for navigation are ultrasonic sensors. With ultrasonic sensors facing the front of the OTV we can program it to differentiate the height difference between the walls from the obstacles, and the log. Since the obstacles are taller than the height of the wall, we can program the sensors to identify and distinguish the wall and obstacle. By doing so, we can orient the OTV to face the correct direction. Initially we were conflicted on whether we should use either ultrasonic or infrared sensors, but ultimately settled on ultrasonic sensors. Ultrasonic sensors have a more accurate and reliable reading in determining distance compared to infrared sensors.

There are a number of challenges associated with programming our OTV to navigate the arena with the aid of sensors and the eye in the sky. The first major one is being able to take the output of the sensors in volts and convert this to a distance, which we can use to see how far the OTV is from an obstacle or wall. This can be done by observing the output of the sensor at different known ranges and creating an equation which can be used by the arduino to convert to our desired unit of distance. Another challenge is telling the robot how to react in different situations. A purely linear approach to coding will not work. We need some sort of function to get the robot 'unstuck' or 'reset' it to a known position and continue with navigation from there. These are just two major challenges but there are many others associated with navigation such as creating functions to reliably turn the OTV a desired amount.

There are two physical characteristics which are important for the navigation team. One of these is the use of ultrasonic sensors. This requirement is shared by the mission and navigation teams. The specific placement of these is determined more by the mission team who have a far more specialized need for infrared sensors. As long as there is one sensor facing forward, which can be used to find distance, it can also be used for navigation. Another requirement navigation has is a narrow chassis. As stated before, the goal is to allow the OTV to be narrow enough to fit in between the wall and any obstacles. We already know the closest any obstacle may be to the wall is 250mm, meaning if we have a 15mm cushion on either side, the OTV would ideally be less than 220mm wide. The reason we think this is feasible is because as the crash site mission, our OTV will not need to physically interact with or move any objects. This means there is no size requirement to complete the mission and we can make the OTV as small as we can fit our components in. However, even with this reduced size, there is still not a lot of room for error to fit the OTV between the obstacles and the wall, meaning it will be challenging to work to make our steering as accurate as possible to fit the small space.

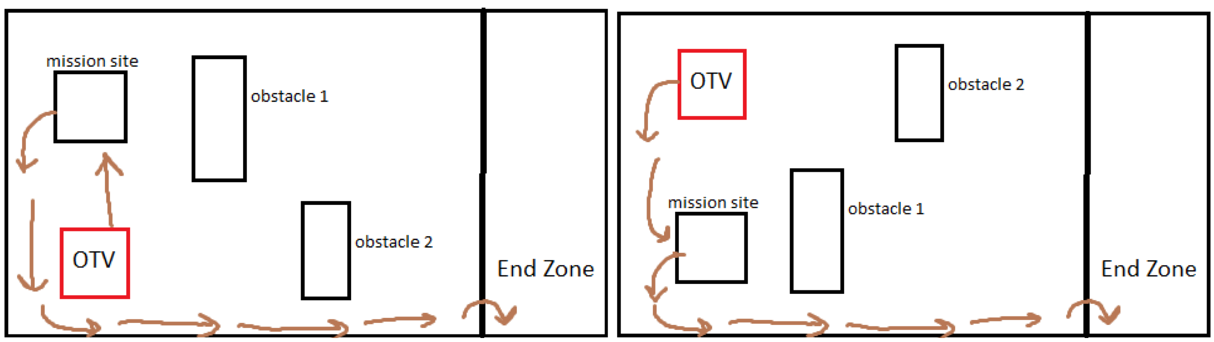
Some issues or edge cases we may run into include the OTV getting stuck. While the chassis is supposed to be narrow enough to fit in the gap between obstacles and the wall, there could be other errors in our timing or code which mean the OTV cannot make the gap. This is what the “stuck” function is for. It is meant to bring the OTV up and around an obstacle before it continues on its way. If the “stuck” function does not work, we run into a real problem. The function could be run again, or there could be another secondary function which takes more drastic measures to ensure the OTV is unstuck and can continue on its way. We need to envision scenarios and perform tests to determine which secondary fix will work better for our OTV.

Navigation Flowchart



While this is happening, there will be a loop constantly reading output from the sensors to determine the distance from the wall. This distance is what will trigger the different turns and steps in the flowchart. The OTV will know to run the stuck function if its position does not change for an extended period of time. This will also be determined by a loop, but this one will be taking outputs from the eye in the sky and analyzing those.

Possible Paths



[6] Diagram of Possible Paths the OTV Can Take

These show just two possible paths the OTV may take. The purpose is to convey the idea of having the OTV drive around the perimeter of the arena to make it to the end zone.

OTV Mission

OTV Mission Requirements

The mission for the crash site involves a rectangular prism covered on all sides with metal panels. One of the panels of metal will be removed from a side, making this side “abnormal”. The box will be placed on a random side, such that the x, y, and z dimensions of the box will not be known until the robot is sent to measure them. To successfully complete the crash site mission objectives, the OTV must first navigate to ± 150 mm of the wreckage. It should determine which side of the wreckage is abnormal, and transmit the direction of the abnormality to the vision system. It should also place the Aruco marker such that it is within 20 degrees of the correct direction. The OTV should also be able to find the dimensions of the abnormal side of the wreckage. These dimensions should be accurate to within ± 25 mm.

Justification

We decided to use ultrasonic sensors to measure the dimensions of the box because it is less affected by outside factors such as ambient light and dust. Ultrasonic sensors also generally have better range than sensors that rely on light. Additionally, using ultrasonic sensors is efficient because they will cover the navigation needs of the robot. In order to ensure the ultrasonic sensor measuring the height of the wreckage box would be high enough to account for all possible heights of the box, we decided to implement a pole which the sensor would be placed upon facing towards the ground. This way, the pole would allow the sensor to be at a height greater than the longest dimension of the box. We decided to use reflectance sensors because the abnormal side of the box will have a panel removed, meaning that part of the side will be black. Shining a light on the black side will yield a different amount of reflected light than from a shiny metal surface.

Mission Structure/Apparatus

For our mission, we will use two ultrasonic sensors along with two infrared sensors to complete our mission tasks. The ultrasonic sensors are to be used to determine the dimensions of the abnormal side, and the infrared sensors are used to determine the abnormal side of the wreckage. Our first ultrasonic sensor will be mounted on top of our chassis base with height $6 < \boldsymbol{h} < 12$ centimeters and will be oriented the same direction as the front of the vehicle. Our second ultrasonic sensor is placed on our pole with height $\boldsymbol{h} > 28$ centimeters facing downwards (-z). Both ultrasonic sensors will be placed at the same location on the chassis, with the only difference being their height. One will be placed at $\boldsymbol{h} = 6$ cm and one will be placed at $\boldsymbol{h} = 16$cm and both will be oriented the same direction as the front of the vehicle.

Sensor Details

<u>Ultrasonic Sensors</u>

The ultrasonic sensors our team decided to use are the HC-SR04 ultrasonic sensor, which were in our kits. These sensors have 4 pins: VCC, GND, ECHO, and TRIG. The GND and VCC should connect to ground and the voltage respectively. ECHO and TRIG both require a digital pin on the Arduino. This means that a total of 4 digital pins will be used for the ultrasonic sensors.

The HC-SR04 specifications are as follows:

- Resolution: 0.3 cm
- Range: 2 - 450 cm

Infrared Sensors

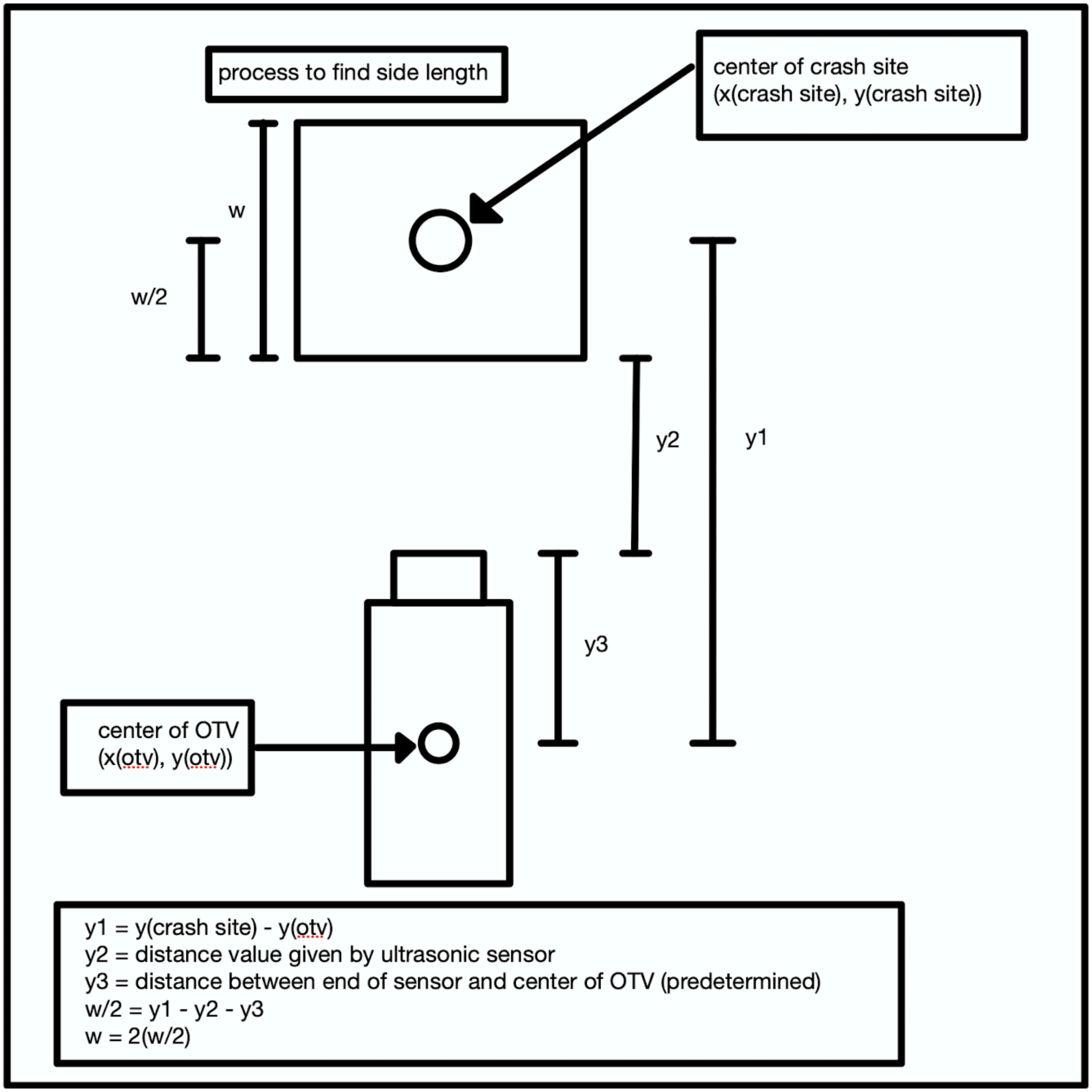
QTR-HD-02RC

The sensors our team decided to use for determining the abnormality are infrared sensors that measure reflectivity of an object. These sensors have an LED and phototransistor pair that allows it to measure the reflectance of an object. The sensor is analog, so the sensor will take up analog pins on the Arduino board. Each sensor will need 3 analog pins (for control, channel 1 and channel 2), for a total of 6 analog pins used on the board.

The specifications are as follows:

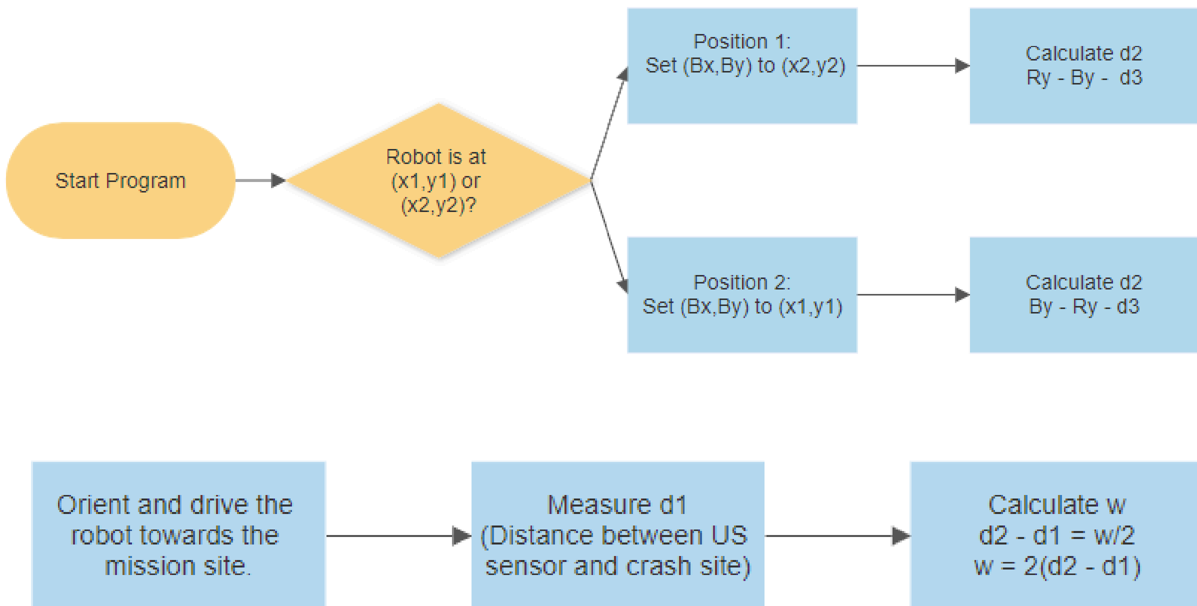
- We could not find a value on the manufacturer's website about the resolution. However, the website did say that the output was an analog value from 0 V to VCC.
- Optimal range: 5 mm
- Max range: 30 mm

Width (Background)

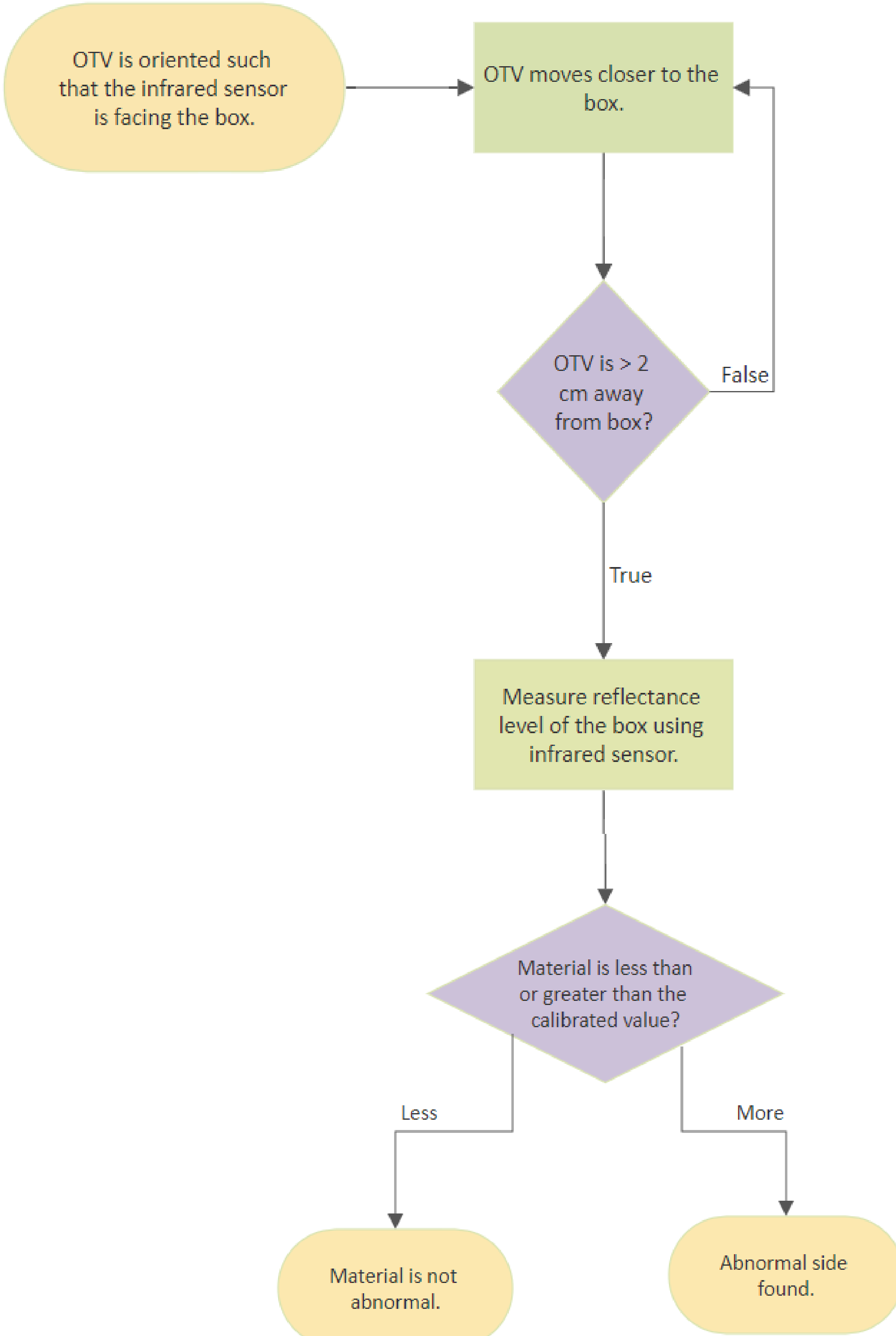


[7] Diagram for determining the width of the wreckage piece

[8] *Flowchart: Width*



[9] *Flowchart: Determining Abnormality*



Systems Integration Plan, Risk Analysis, Pin Assignments

Systems Integration Plan

Mission will collaborate with Structure in order to attach required sensors onto the chassis. Locomotion and Structure will collaborate to make sure the wheels can rotate properly within the chassis. Locomotion and Navigation will work together to figure out the needed rolling resistance values to overcome the log. Mission and Navigation will work together to make sure sensors can be used to achieve both teams' goals.

Risk Analysis

Risk 1 (locomotion): Chosen materials may not be enough to overcome rolling resistance to go over the log.

- Risk level: low.
- Solution: Identify problems early and figure out better materials to fix problems.

Risk 2 (navigation): Coordinate system given by bird's eye might not be accurate enough.

- Risk level: medium.
- Solution: Implement a different coding strategy to alleviate the wrong/missing coordinates.

Risk 3 (mission): The pole is too heavy or high so that the OTV may flip over the log when attempting to climb.

- Risk level: high.
- Solution: Change pole material, implement a different design to move the center of mass.

Risk 4 (structure): Chassis gets stuck on the side of the wall while maneuvering the obstacles.

- Risk level: medium,
- Solution: Reevaluate chassis shape, find a different strategy to avoid obstacles besides hugging the wall.

Pin Assignments

Analog:

A0 Infrared Sensor 1

A1 Infrared Sensor 1

A2 Infrared Sensor 1

A3 Infrared Sensor 2

A4 Infrared Sensor 2

A5 Infrared Sensor 2

Digital:

D0 Ultrasonic Sensor #1

D1 Ultrasonic Sensor #1

D2 Ultrasonic Sensor #2

D3 ～Motor Controller #1

D4 Ultrasonic Sensor #2

D5 ～Motor Controller #1

D6～Motor Controller #1

D7 --

D8 --

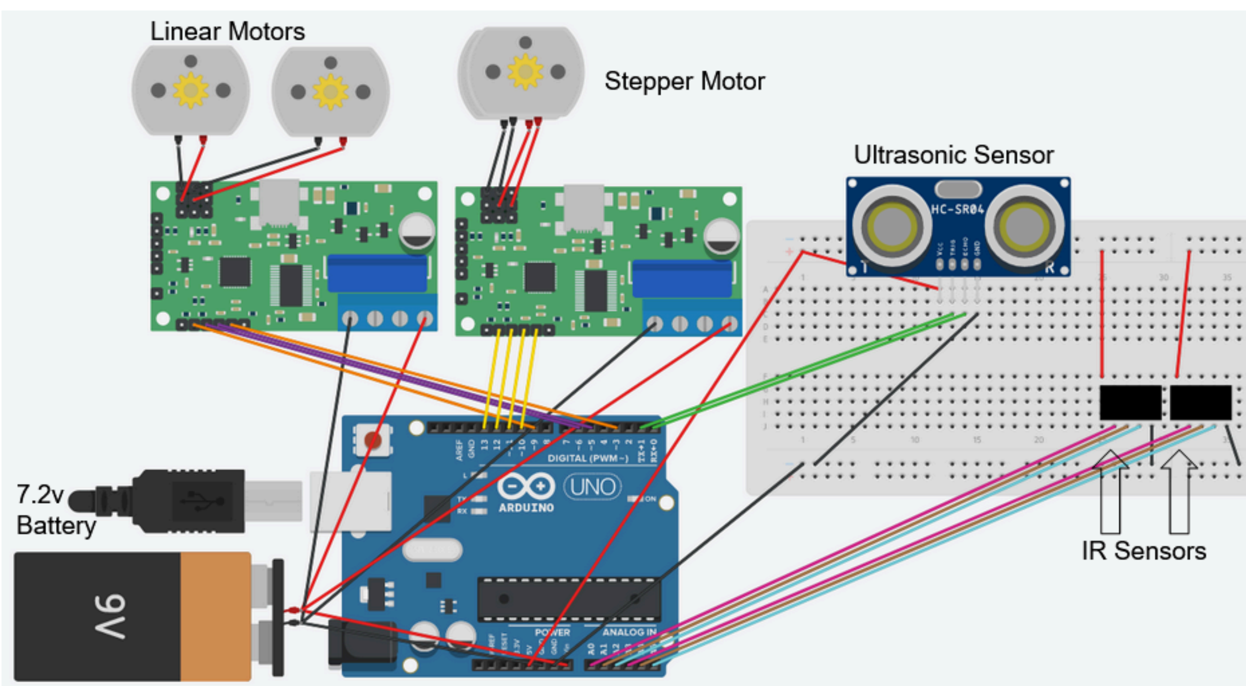
D9 ～Motor Controller #1

D10 Motor Controller #2

D11 Motor Controller #2

D12 Motor Controller #2

D13 Motor Controller #2



[10] Schematic Diagram for OTV

Preliminary Bill of Materials

Item	Mass (g)	Approval	Quantity	Cost
Arduino Ultrasonic Sensor (in kit)	2	Yes	2	
QTR-HD-02A	0.6	Pending	2	$2.12
Motor	47	Yes	2	$21.95
Stepper Motor	180	Pending	1	$17.95
Battery + Charger	125	Yes	1	$20.49
Wheels (pack of 4)	266.4	Yes	1	$23.90
Motor Controller	50	Yes	2	$8.99
Extra material costs (wood, metal, 3d printing)	233.5	Yes		
Total	952.1			$128.46

[11] Bill of Materials

Teamwork

How does this design reflect the team's goals and interests?

The design is a result of each subteams' work and collaboration between subteams to make sure all parts of the OTV work together while satisfying the goal for each subteam.

How is the team drawing on individual strengths?

Team members with more experience in a specific aspect of the OTV / mission will guide the team. Our team members are split into separate subcategories where they feel more competent and comfortable. For example, members who have more extensive experience with CAD will focus on the structure portion and members with more experience with computers and sensors will focus on the mission and navigation portion.

How is the team advancing individual growth?

Each subteam deals with specific problems that they will have to overcome, learning something in the process. Every member gains more experience and knowledge about engineering, coding, and the Arduino.

Citations

Ultrasonic Sensor HC-SR04 with Arduino Tutorial. (n.d.). Arduino Project Hub. https://create.arduino.cc/projecthub/abdularbi17/ultrasonic-sensor-hc-sr04-with-arduino-tutorial-327ff6