Engineering Journal

Team Null

Contents

Introduction

Welcome to Team Null's engineering journal. Team Null is made up of Michael Shepstone (19 years), Raaid (18 years) , our coach Cody Williams and Gremlin, our vehicle's vehicle. We are from South Africa. As a team we have been working on the Future Engineering challenge for the past 2 years. Gremlin, the vehicle, has evolved many times to meet the challenging needs of the Future Engineers Category of the World vehicles Olympiad 2024. This document is relevant to the current latest version of Gremlin, Gremlin V6.

1. Mobility Management

Introduction

The chassis of the vehicle provided a unique challenge. A number of valuable lessons were learnt from the vehicles' performance in the 2023 National Finals, and as a team it was determined that further versions of this vehicle must have a far tighter turning circle, as well as the ability to reverse, in order to successfully navigate the obstacle challenge course. Additionally, it was further determined that it was necessary to make the vehicle as compact as possible in order to provide the maximum margin of error when navigating the obstacle challenge course.

Motor

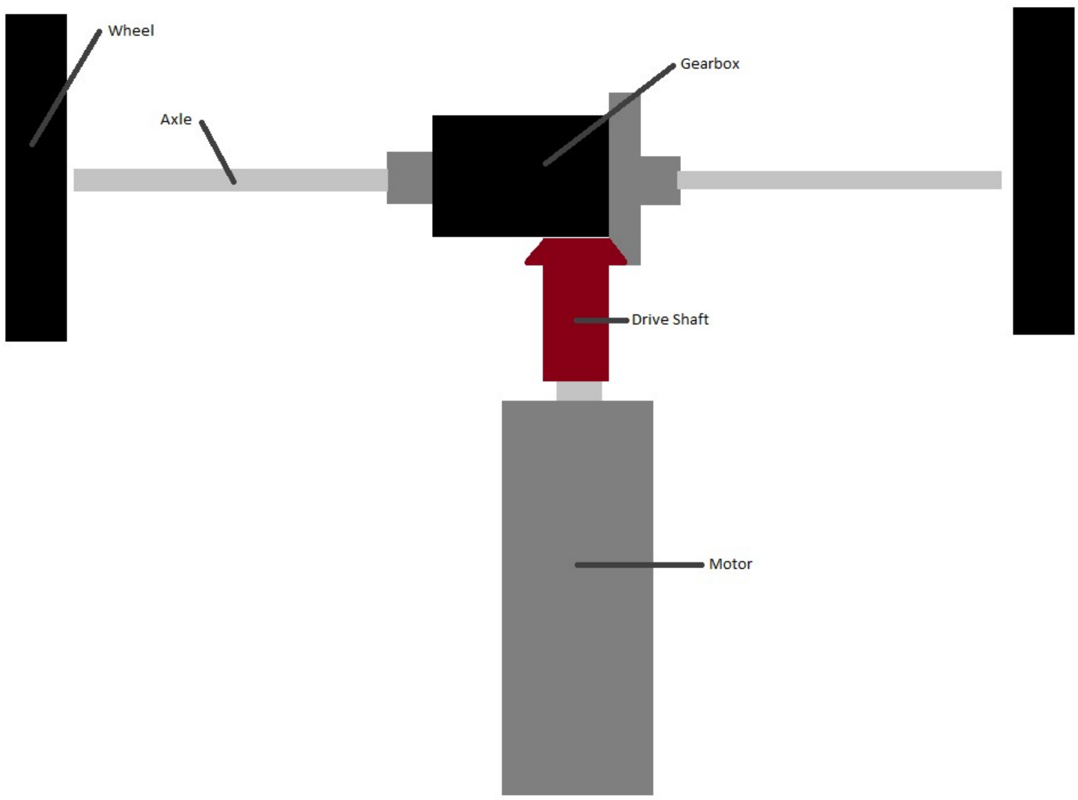
A single 6W 12V brushed DC motor with an integrated gearbox was used. This motor produced 282 RPM at the output shaft. Only the rear wheels are driven by the motor. There was a period of time where this motor was exchanged for another similar 6W 12V DC motor, this motor has a higher RPM output, of around 610 RPM at the output shaft. Unfortunately, this faster motor resulted in the steering control loop becoming inadequate due to the relatively slow response rate of the steering loop (~250ms) resulted in the vehicle veering of course before the steering could be corrected. After experimenting with different steering methods, and motor speeds (slowing down the faster motor resulted in poor torque output), the decision was made to switch back to the original motor for the obstacle challenge, where the precision was required, and use the faster motor for the open challenge, where the precision is not required. Although this would require a motor change between rounds, it was the best solution that we could reach with the tools on hand.

Powertrain

Only the rear set of wheels are powered by a single driving axle system.

The motor, mounted parallel to the direction of movement of the vehicle, is attached to a custom, 3d printed shaft. This shaft adapts the output of the motor to the input of the gear system. This gear system, identical in concept to those found on production cars, translates the rotation of the motor shaft by 90 degrees to the driving axle. Additionally, this gearbox reduces the rotation speed at the approximate ratio of 4:1.

The rotation is then translated through a telescopic shaft and two constant-velocity joints, necessary to allow the wheel to steer and receive power, before arriving at the wheel. A simplified diagram of this system is shown below



Steering

Achieving a far smaller turning circle than previous versions of the vehicle was a top priority for this competition. It was therefore decided to implement an all-wheel steering system, inspired by that of the “Tesla Cybertruck”. This system uses a series of gears and linkages in order to invert the motion of the steering motor and transfer its power to the rear wheels. This allows the single steering motor to actuate both the front and rear wheels, in opposite directions, at the same time, dramatically reducing the turning radius of the vehicle. To further improve the steering performance of the vehicle, the wheelbase was kept to a minimum.

The entire steering system is actuated by a single 14KG MG996R metal gear servo. Initially, 9g servos were used for steering, but these would burn out quickly under the increased loads placed upon them by the steering mechanism. It was therefore decided that a more powerful (metal gear) servo was necessary, although more powerful servos can be used and will work, not only is this extra power unnecessary, but can potentially damage the steering system should an erroneous movement be made.

Power and Sense Management

Power Management

The vehicle contains two major power systems, these systems are connected by a common ground to allow for data transfer between them.

The first is powered by four 3000mah Samsung 18650 battery cells. This is then fed into 4 3A adjustable buck convertors to feed different voltages to the servo, Raspberry Pi Picos and dive motor. This system is continuously monitored by a voltmeter in order to preserve the lifespan of the batteries.

The 2nd system is powered by two 3000mah Samsung 18650 battery cells. This is fed into a single 5A adjustable buck convertor. This system is used to exclusively power the Raspberry Pi 4 and the webcam.

The reason for the separation of the larger system into two separate systems is a result of the combination of current draw spikes between the drive and servo motors causing a voltage drop to the Raspberry Pi 4. This would cause the board, which appears to be highly sensitive to input voltage, to reboot (a process that takes about 50 seconds) mid run, ruining the overall run. This would seemingly happen at random, and although was fairly uncommon (this issue would occur once every 4 runs) was enough of an issue to warrant the separation of power systems. Although it could possibly be solved with circuitry, in the interest of time, effort and simplicity, it was decided to split the systems.

Sense Management

Sensors included

1. 5 x TF-Luna Time Of Flight (TOF)
2. 1 x Generic webcam (sourced from a team member's desktop computer)
3. 1 x Adafruit BM085 gyroscope

Mounting Positions.

The TF-Lunas are positioned on the skirt of the vehicle. Three of the sensors are halfway down the skirt, namely the front, 45 degrees left and 45 degrees right sensors. These are angled parallel to the ground. The 90 degrees left and 90 degrees right sensors are mounted to the top frame, near the front of the vehicle. Due to clearance issues with the wheels, they had to be mounted higher up than the other sensors. As a result of this higher mounting position, they had to be angled downwards by approximately 2 degrees to ensure that they would never look over the walls of the track.

The webcam is glued in the front of the vehicle, attached to the top frame, facing forward.

The gyroscope is mounted to a protoboard as a part of the steering system. This sits towards the back of the vehicle on top of the upper frame.

Vehicle Control

There are 4 microcontrollers and 1 Single Board Computer (SBC) to control the car:

1. 1 x Pi Pico for managing the drive motor
2. 1 x Pi Pico for reading data from the gyroscope and managing the steering servo
3. 1 x Pi Pico that acts as the "main" controller
4. 1 x Pi Pico for facilitating I2C communication between the SBC and the main controller
5. 1 x Pi 4 that runs a python script leveraging OpenCV to detect obstacles

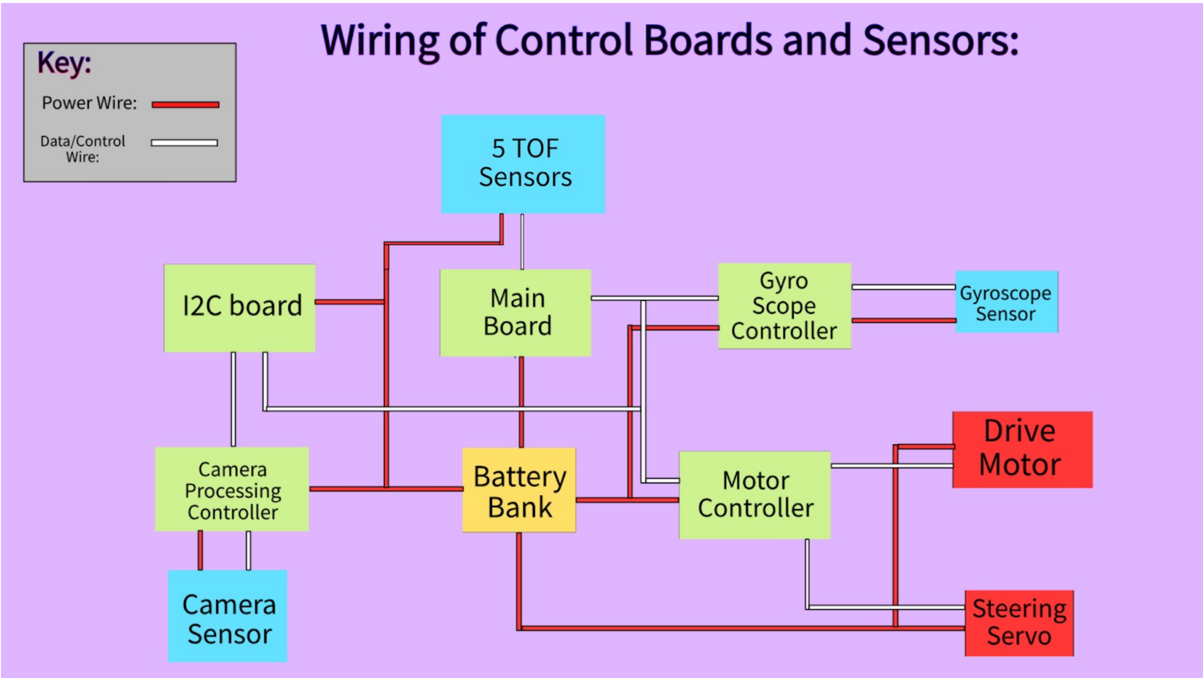


Figure 1: Wiring of Control Boards and Sensors

Each Pi Pico has two I2C buses. The first bus is used to communicate with the I2C devices that a particular board is responsible for managing (if any). The 2nd bus is used for communication between boards. This is with the exception of the “I2C board” which uses both buses for inter-board communication for reasons that will be explained later.

Each board has a particular purpose. Although it is possible to have one board serve multiple purposes, in the interest of software simplicity, modularity and avoiding issues associated with the limited processing power of the pi pico board (The pi 4 was only added after a large amount of work was done with the Pi Picos).

The main board is responsible for the major decision making. It handles all the TF-Luna sensors as well as the overall program logic. It processes data from, and issues command to, all the other boards connected to the secondary bus. It is the master on both I2C buses.

The gyro board is responsible for handling the steering feedback loop of the vehicle. On the primary bus is the BMO 085 gyroscope sensor, and an OLED display. On the secondary bus, it can receive a new target angle from the main board, and report the true angle to the main board. It is the master on the primary bus and a slave on the secondary bus. This board constantly polls the gyroscope and makes adjustments to the servo motor in order to keep the vehicle on course.

The motor board is responsible for controlling the speed and direction of the main drive motor. The primary I2C bus has no connections and the secondary I2C bus is used to receive motor speed commands from the main board. It is a slave on the secondary bus.

The camera board (pi 4) is responsible for tracking the position, size and colour of objects of interest. It is constantly reporting this information on its primary I2C bus. It is the master on this bus.

The I2C board acts as a middleman between the main board and the camera board. Due to the kernel of the Raspberry Pi 4 not allowing it to be a slave on an I2C bus, and the architecture of the vehicle already dictating that the master board by the time that this discovery was made, this board was made necessary. The camera board constantly reports the position, colour and size of objects of interest. This board caches these reports, making them available for the main board should the I2C request be made. Although it was initially thought that this system would add too much latency, this setup works surprisingly well for this application. This board is a slave on both the primary and secondary bus.

Obstacle Management

The TOF (Time of Flight) sensors are used to detect the distance from the boundary walls to the left, right and front. This data is used in combination with the gyroscope to navigate around the course.

Open Challenge

During the open challenge round, only these sensors are used to navigate the course. Although the camera is actively reporting data to the I2C cache board, that data is never read by the main board.

During the initial movement, data is read from both the left and right TF-Luna sensors in order to determine the correct direction to turn.

Once the turning direction is established, only the inside sensor is polled, waiting for the wall to fall away before a turn is initiated.

After a turn, the vehicle will wait until it has acquired the target angle (90 degrees from the angle it was targeting before the turn. The vehicle begins the course targeting 0 degrees). This is necessary as the TF-Luna sensors cannot measure the distance from a wall should they be more than 5 degrees away from the perpendicular to that wall. Once the vehicle has reached its target angle, it will ensure that the wall is present before waiting for the wall to fall away. This is to ensure that the vehicle cannot make two “turns” in rapid succession, before it has reached the next inside wall, resulting in it doubling back on the path it came from, and inevitably getting stuck against a wall somewhere.

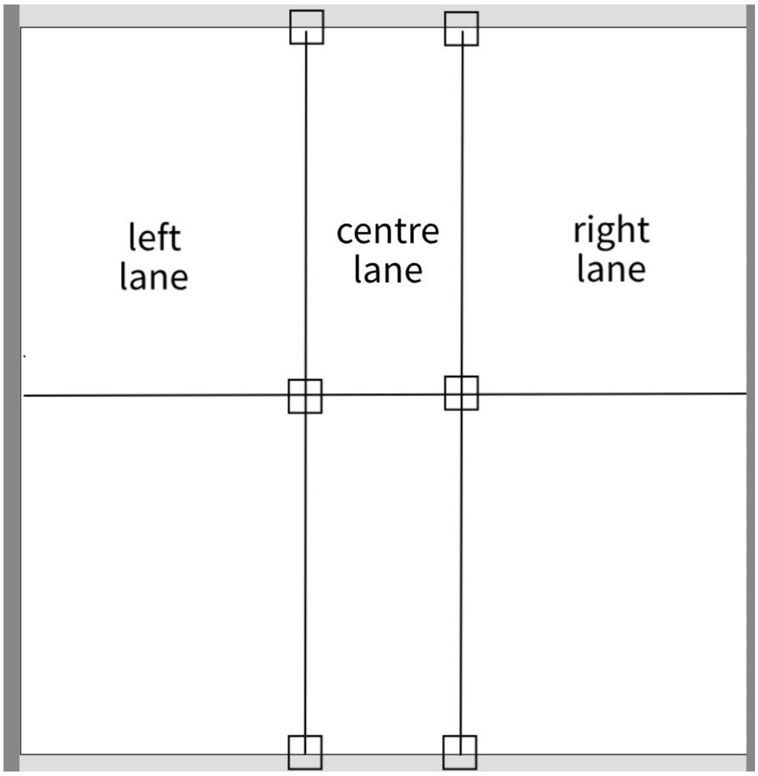
Obstacle Challenge

During the obstacle challenge, the vehicle performs all the wall operations from the open challenge with a few extra steps.

The avoidance of obstacles is divided into two main sections, avoidance of obstacles while the vehicle is travelling in a straight section, and avoidance of obstacles while the vehicle is turning.

The course is divided into three logical lanes, the left, right and centre lanes. The left lane is as close to the left wall as possible, and vice versa to the right. The centre lane is slightly biassed towards the outside lane, be it left or right, depending on the round direction. This is to facilitate the reuse of code between the lack of object and travelling in the outside lane in the presence of the parking block.

While the robot is navigating the course, it is building a map of it. This increases the reliability of obstacle handling as well as allows for the location of the parking area.



Obstacle Avoidance in Straight sections

Just after the first obstacle has passed, a frame is captured from the cache. Based on this frame, the vehicle will make the decision to navigate to the left or right lane.

> Should the lane the vehicle is currently in be the lane it needs to navigate to, no navigation action will be made. Should a lane change be necessary, the vehicle will angle itself at 45 degrees, and use the appropriate TF-Luna sensor to measure its distance from the wall it is approaching. This will allow it to successfully navigate to and hug the wall on the side of the course as dictated by the colour of the block observed.
>
> Should no block be observed, the vehicle will remain in the lane it is currently in
>
> Should the parking area be observed, the vehicle will make no adjustment if the target lane is the inside lane (whether this is the left or right lane depends on the direction the vehicle is travelling). If the vehicle is targeting the right lane, or is currently inside the right lane, the vehicle will target the centre lane instead.

Obstacle Avoidance in Turns

The position in which the frame is captured depends on the vehicle position. Should the vehicle be in the inside lane, the frame will be captured before the turn is initiated. Should the vehicle be in the outside lane, the vehicle will turn 60 degrees before capturing a frame. Based on the contents of this frame, the vehicle will make one of six unique movements depending on the current and target lanes of the vehicle.

Due to the method of turning, while originating from the inside lane, the vehicle will never hit the parking area, due to it basing its turn off of the position of the block should the outside lane be targeted. This naturally puts it into the centre lane, should the parking block be present.

Should there be no block present, and the robot originates in the inside lane, the robot will use its front TF-Luna to determine the correct point at which to turn.

Should the parking be present, and the robot originates from the outside lane, the centre lane will be targeted instead of the outside lane. The move action is then identical to the action should there be no block.

Parking.

When navigating to the parking area. The robot will hug the inside wall until it reaches the segment where the parking is located. The robot will then navigate to the centre lane and use the outside TF-Luna sensor to determine the location of the parking area. It will then swing into it directly and stop the motor after 3 seconds. Due to the previously built map of the course, the camera is not needed for this operation.

Frame Assembly

Bill Of Materials:

Material	**Quantity**
Major Mechanical Components	
PLA Filament	Approx 500g
56mm tyres	4
JGA25-370 Geared motor DC motor 12V electric gear motor high torque 282 rpm	1
Metal Differential And Drive Shaft Set For SCY 16101 16102 16103 16201 Pro 1/16 Brushless RC Car Upgrades Parts Accessories	1
MG996R (180 degrees) Servo Motor	1
6.125mm-12mm single channel ball bearing	4
Minor Mechanical Components	
20mm M2.5 standoff	4
6mm M2.5 bolt	4
5mm M2.5 standoff	4
5mm M3 standoff	4
10mm M3 standoff	2
M3 Nylon nut	2
M3 standard steel nut	9
M3 lock-nut	8
6mm M3 threaded Inserts	8
6mm M3 bolt	10
8mm M3 bolt	4
10mm M3 bolt	2
12mm M3 bolt	5
16mm M3 bolt	4

25mm countersunk M3 bolt	1
8mm M1.25 bolt	6
M1.25 standard nut	6
Major Electrical Components	
Raspberry Pi Pico	4
Raspberry Pi 4	1
Wide Angle 1080p Webcam	1
TF-Luna Time of Flight Sensor	5
Adafruit 9-DOF Orientation IMU Fusion Breakout	1
64x32 I2C OLED Display	1
18650 Battery Cell	6
2x 18650 battery holder case	3
3A Adjustable buck convertor	4
5A Adjustable buck convertor	1
5V relay module	4
LR7843 Isolated FET Module	1
4x3 LED matrix	1
Minor Electrical Components	
JST-XT 6 pin male-female connector set	2
JST-XT 4 pin male connector	7
JST-XT 4 pin female connector	9
JST-SM 3 in male-female connector set	1
JST-Micro 2 pin male-female connector set	1
5mm through-hole LED (Various colours)	19
2x20 2.54mm female header	1
1x40 2.54mm female header	10
1x40 2.54mm male header	10
10kΩ Resistor	5
1MΩ Resistor	1

Through-hole momentary switch	1
Solder	
Wrapping wire	

Printing Instructions

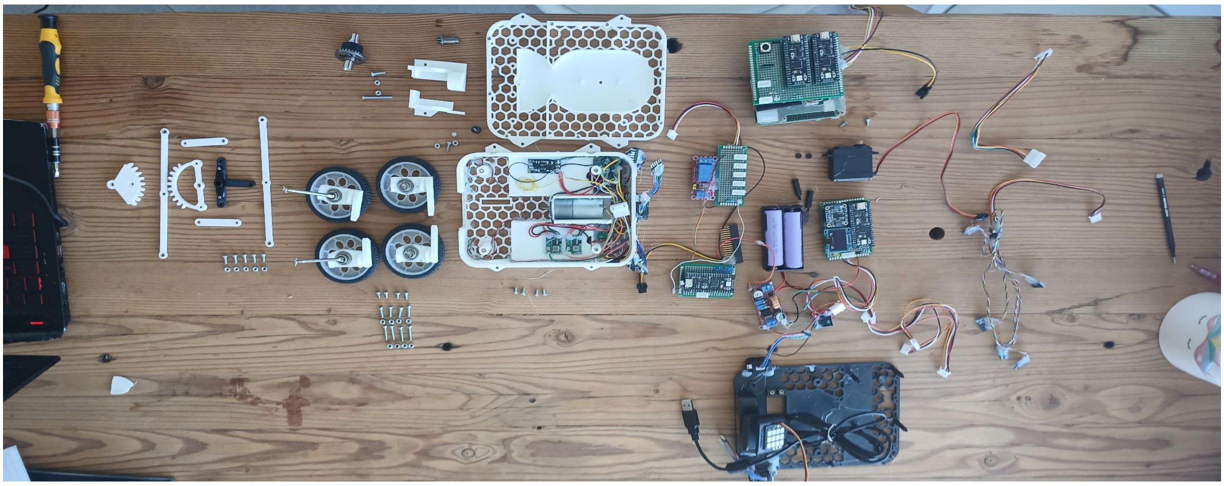
All parts for the vehicle were printed on an Ender 3 V2 Neo 3D printer with a 0.4mm nozzle in standard PLA (Polylactic Acid) filament.

The parts were printed with tree supports, at a layer height of 0.16mm, 0.8mm wall width and a standard infill of 25%. This is with the exception of the printed motor shaft, which was printed with 0.4mm walls and 100% infill. This decision was made after the motor shaft printed with standard settings sheared off, leaving a piece of it embedded in the differential drive gear, which was very tricky to remove.

The files were sliced using Cura Slicer. Should you want to copy the configuration exactly, the Cura configuration is available here.

Some files may have to have their orientation changed away from the default for optimum printing performance. This is usually achieved through rotating the part to align any large flat faces against the bed, thus minimising overhangs.

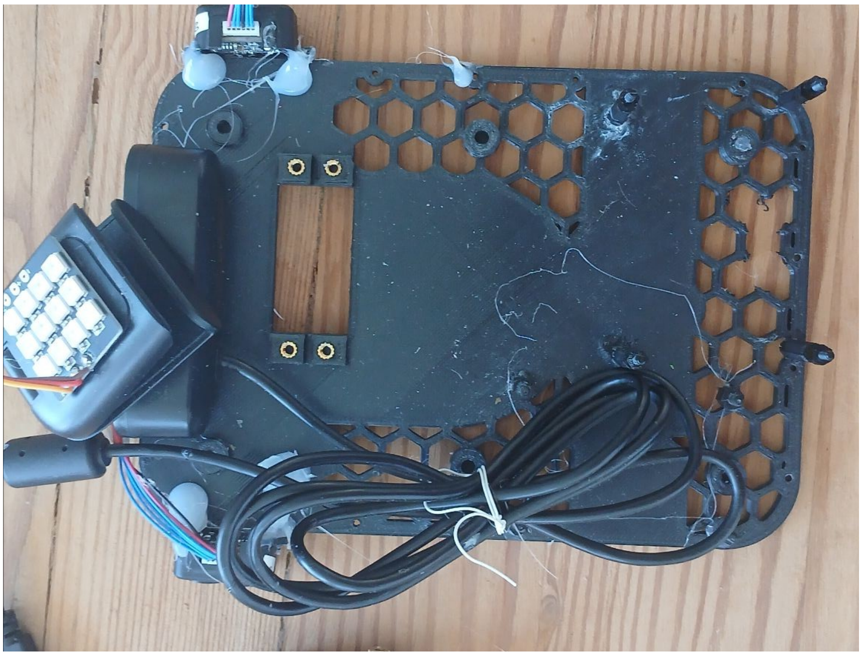
Assembly Instructions



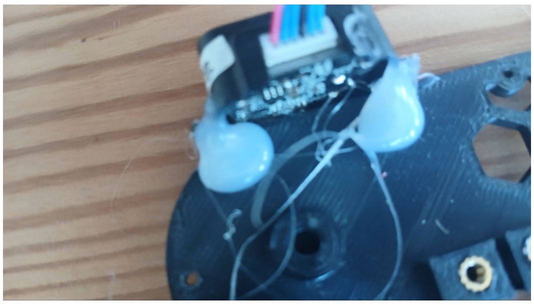
Once electrical components have been assembled as specified by the circuit diagrams (covered later in the document). The assembly of the mechanical components can begin .

Please note, some of the photos show components already mounted, these components have been glued down and could not easily be removed for the purpose of this document. All series of images explaining a process should be read left to right, top to bottom.

Top Frame

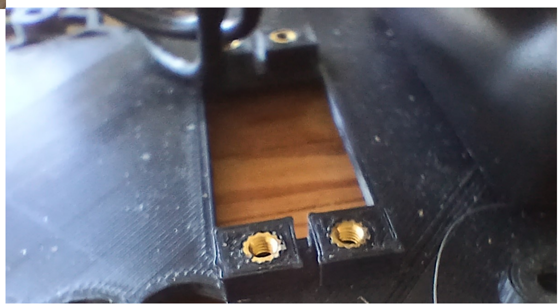


1) Glue the webcam to the 3D printed frame, the same can be done for two TF-Lunas. The TF-Lunas must be angled downwards by approximately 2° to ensure that they do not look over the walls. When these actions are complete, the result should resemble the image above. Below are close-ups of each component in their position:



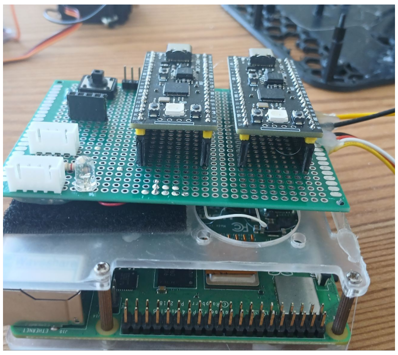
2) Glue the 4x3 LED Matrix to the back of the webcam mount as shown below.

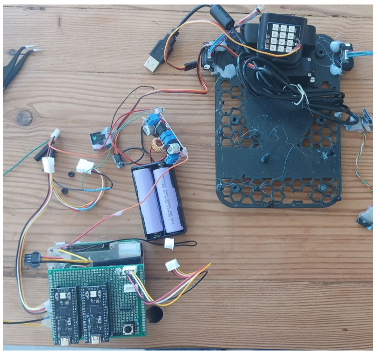
3) Push 4, 6mm M3 threaded inserts into the holes surrounding the servo mount. The result should be as pictured in the image on the right.

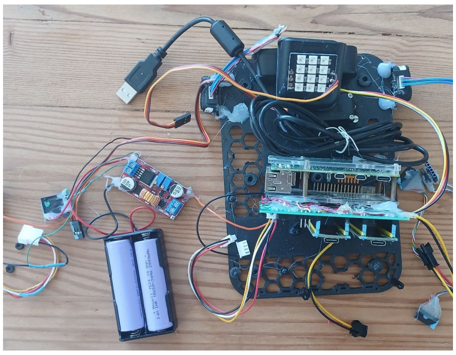


4) Attach the steering servo, the servo should be placed in the hole, and secured with two 6mm M3 bolts. The image below shows this process in progress.

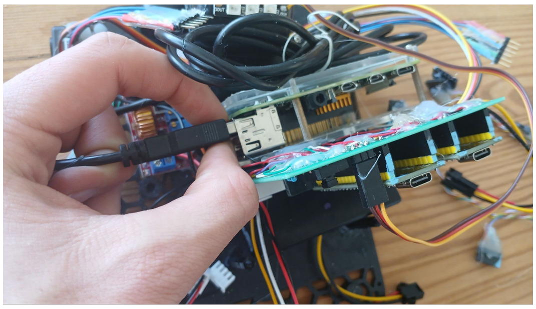
5) Use a small amount of hot glue to attach the Raspberry Pi and main board to the top frame as shown in the images below.

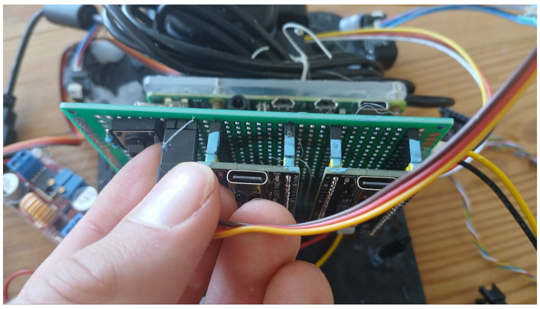




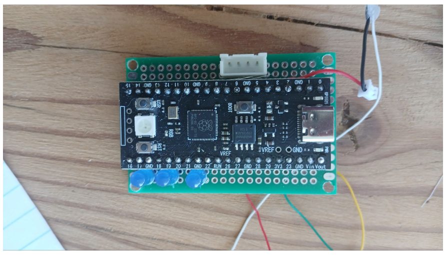
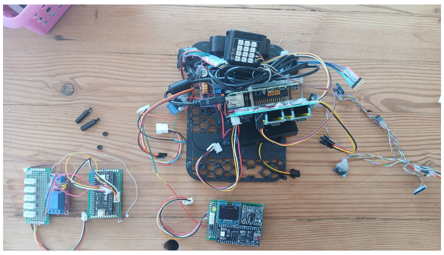
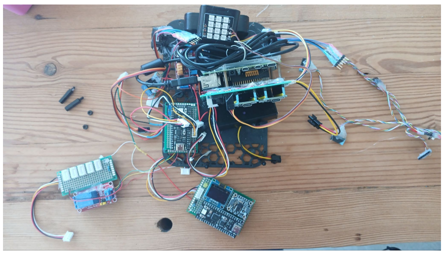


6) Connect the webcam and LED matrix to their relevant ports as shown below.

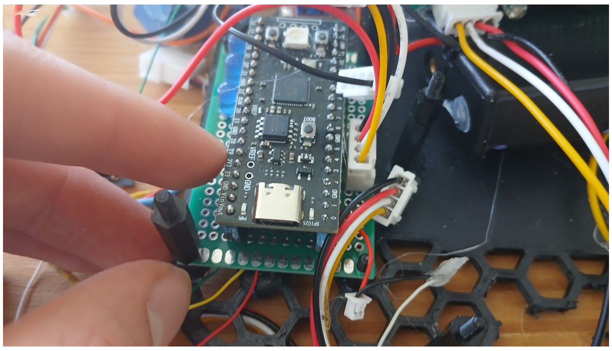
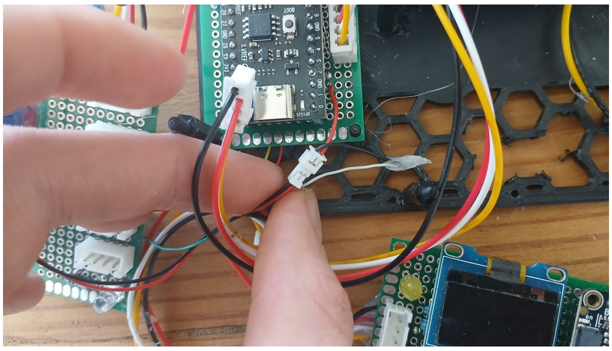
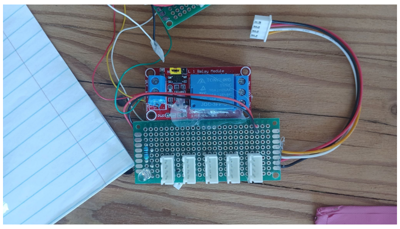
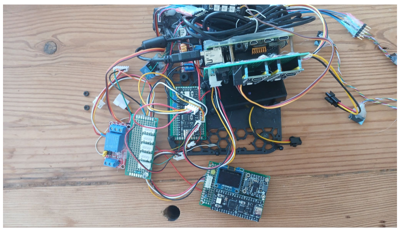
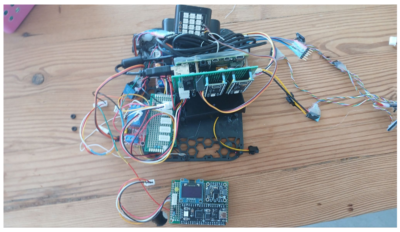




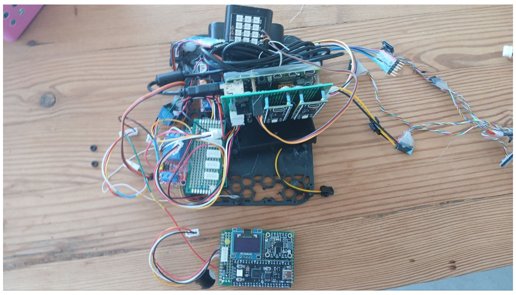
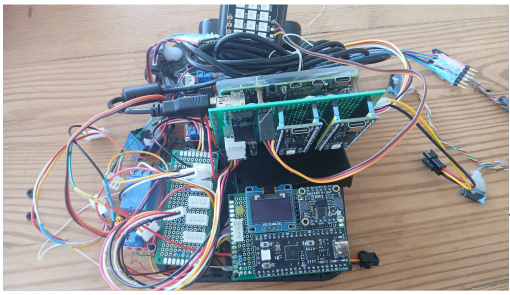
7) Attach the motor control board to 4 5mm M3 standoffs. Place the board onto the frame and glue the standoffs in place as shown in the series of images below.

8) Attach the 10mm M3 standoff to the 5mm M4 standoffs holding the motor control board. This should secure the board in place and provide a mounting point for the wiring hub and motor direction relay. Attach the communication wire (male-male 4 pin JST-XH) to the motor board now, as it will be difficult to reach it later. Attach the wiring hub and motor direction relay to the standoffs above the motor control board. The process is outlined in the series of images below.

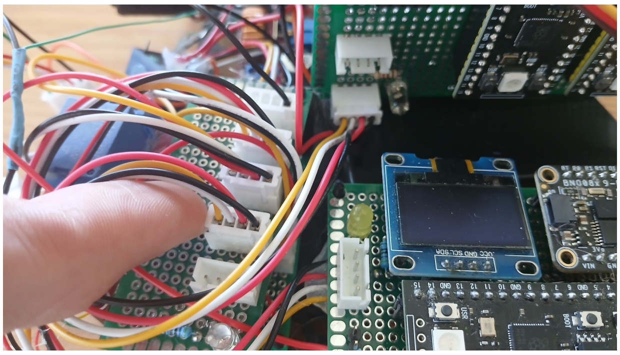
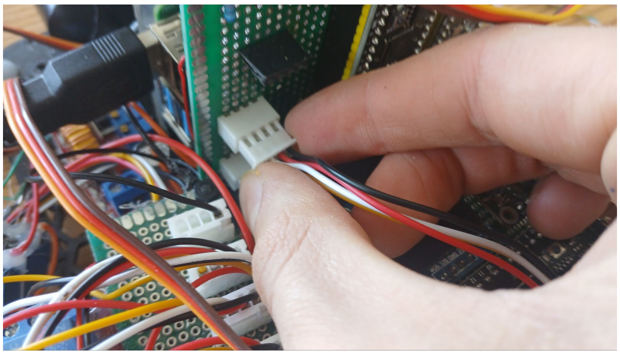
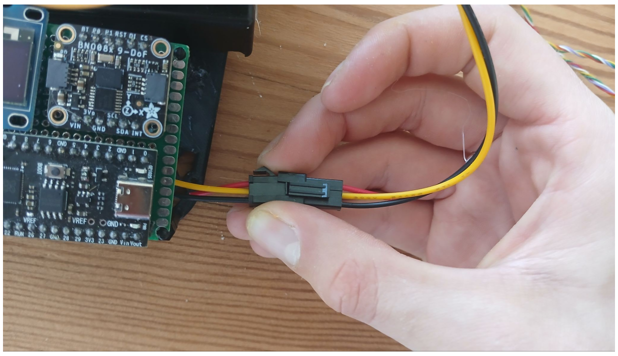
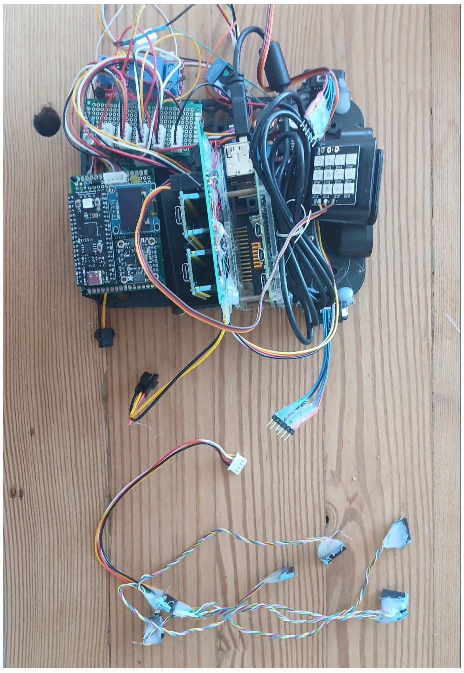
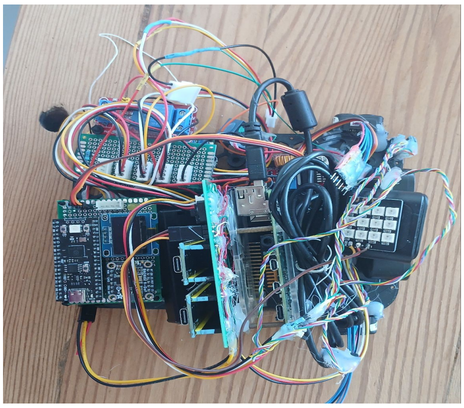
9) Attach the 4 20mm M2.5 standoffs to the 5mm M2.5 standoffs. Place these through the mounting holes on the gyroscope board and secure with the M22.5 standard nuts. Place this entire assembly onto the upper board as shown in the below images and glue the standoffs in place

10) Connect all boards to the wiring hub. Connect the Pi to the main board. Connect the servo to the gyroscope board. Attach the TF-Luna wiring harness to the main board

Lower Frame

1) Mount all the 3A voltage regulators in the frame, as well as two of the 5V relay modules. These should be connected as specified in the wiring diagram. Although it is not necessary, 12 downward facing LEDs were added that are used to indicate the switched on state of the vehicle. Holes were also drilled in the bottom of the frame to provide access to the power switch and a voltage monitor.

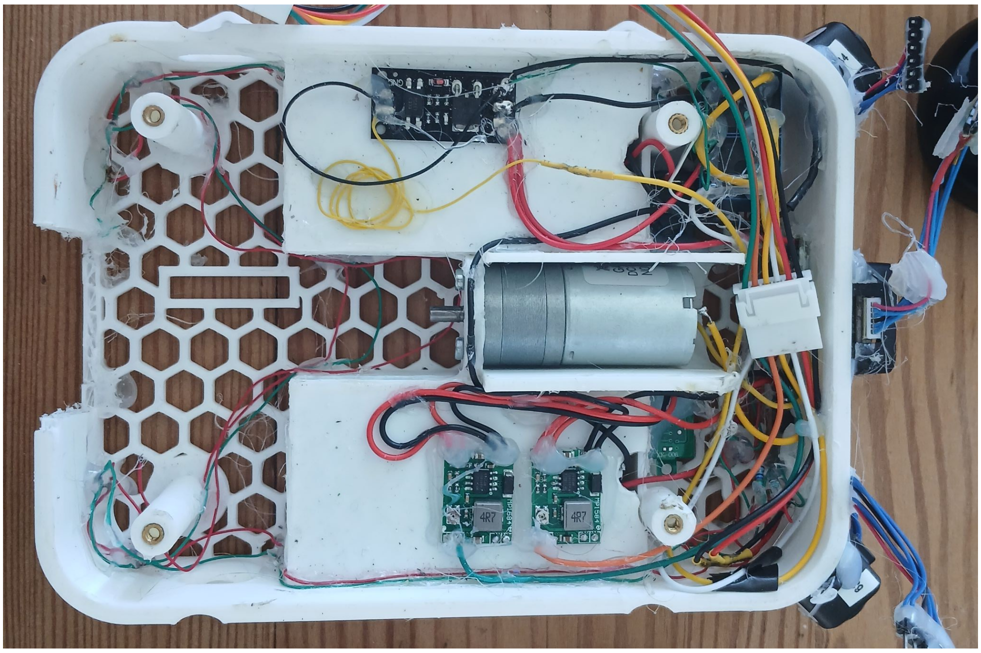






2) The motor must also be attached to the lower frame. This is done by aligning the motor with the boltholes, lowering it into the frame against the back wall before

pushing it forward; it is then secured using two M3 6mm bolts. The motor may have to be rotated slightly in its mount to correct the alignment with the mounting holes.

3) The front facing TF-Lunas need to be mounted. Use a small amount of hot glue to attach a sensor facing directly forwards, left 45 degrees and right 45 degrees. Although the up down direction does not need to be closely monitored (as long as they are aligned with the frame itself, it will be fine) the left-right angle of the two angled sensors needs to be closely monitored as difference in angles between these sensors will cause differences in behaviour between the two directions of the vehicle.

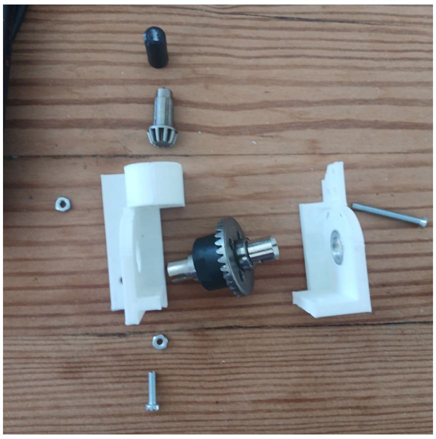
The **final product** is pictured below:

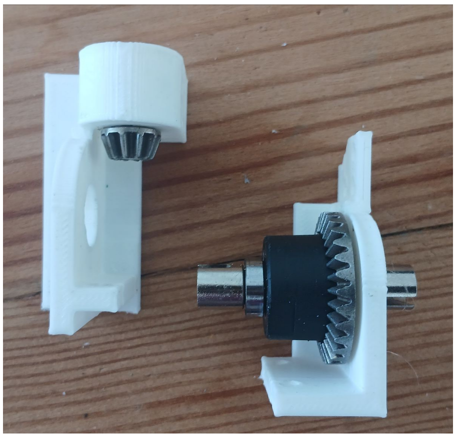


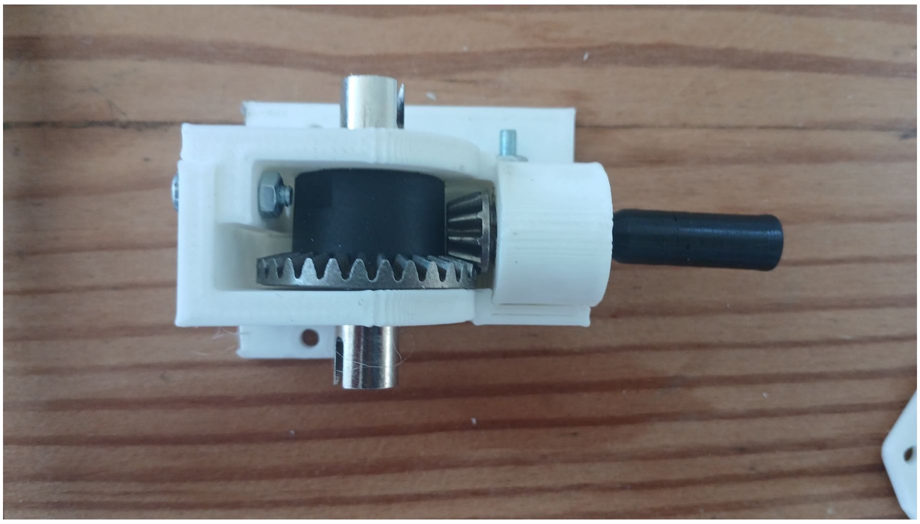
Middle Frame

1) Assemble the wheel subassemblies. The outside surface of the bearings should be coated with two-part epoxy before being inserted into the 3d printed wheel mounts. Care needs to be taken not to get any of this epoxy into the bearing mechanism, which will stop it from rotating. Once the epoxy is dry, the wheel shafts can be inserted into the bearings, the wheels can then be mounted onto the shafts, and secured with the included bolt. This should be repeated for each wheel. Ensure that the back wheel shafts are mounted to the back wheel steering mounts. The image below shows the final product of this process

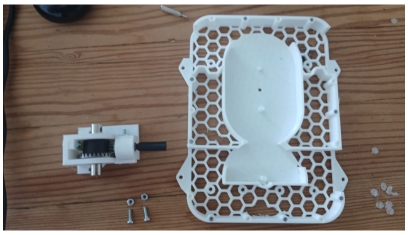
2) Assemble the gearbox frame. Place the gears into the frame as shown in the first two images. Carefully combine the two halves of the gearbox, ensuring that the gears mesh properly. Once the gearbox is combined, thread the 30mm countersunk bolt through the hold below beneath the small gear, before securing it with a nut. The same can be done with the 12mm bolt and the top hold on the left side of the final image. Secure this with a nut, taking care not to drop the nut into the frame, as it can prove difficult to remove. Once the fame has been fully assembled, the motor shaft can be inserted into the hole in the small gear. If the assembly is correct, rotating the motor shaft should cause all the gears to rotate

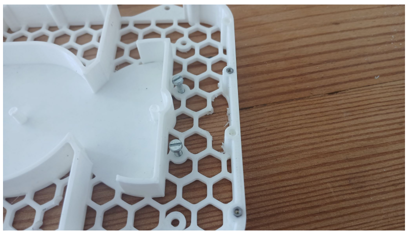


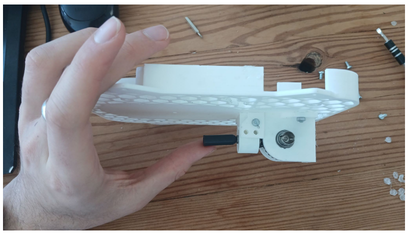




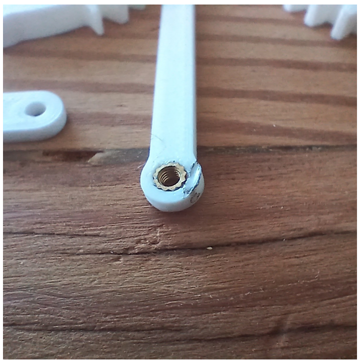
3) Mount the gearbox to the middle frame. Insert two 10mm bolts into the mounting holes as shown in the 2nd image. The gearbox frame should be mounted to the middle frame, with the plate against the middle frame. Once secure with bolts, the final assembly should resemble the 3rd image.



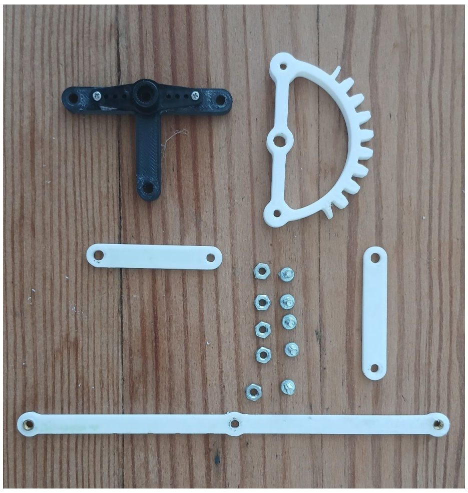


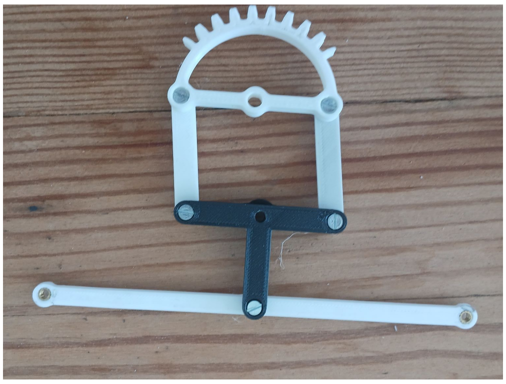


4) Insert M3 threaded inserts into the 4 outer holes of the steering rods. This is pictured on the right.

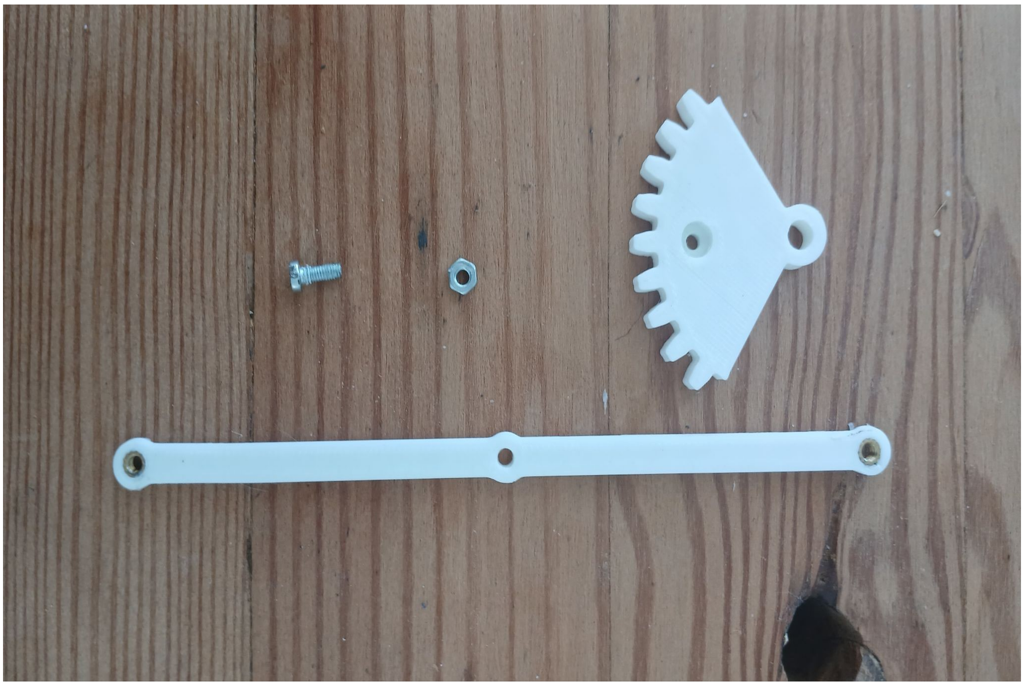


5) Assemble the forward steering mechanism. Glue the servo horn onto the steering actuator (black component in the left image). Place five 8mm M3 bolts into the three countersunk holes on the steering actuator and the two countersunk holes on the gear. The image below left shows the result of this process. Place the spacers over the bolts on the in-line holes on both the gear and steering actuator. Place the steering rod onto the forward bolt on the steering actuator. Secure all the bolts in place using standard M3 nuts. Tighten the bolts until the mechanism can barely hold its shape when held at 90 degrees in the air, do not tighten it beyond this point. Place a small amount of hot glue on the tightened bolts to prevent them coming loose as a result of movement and vibration.



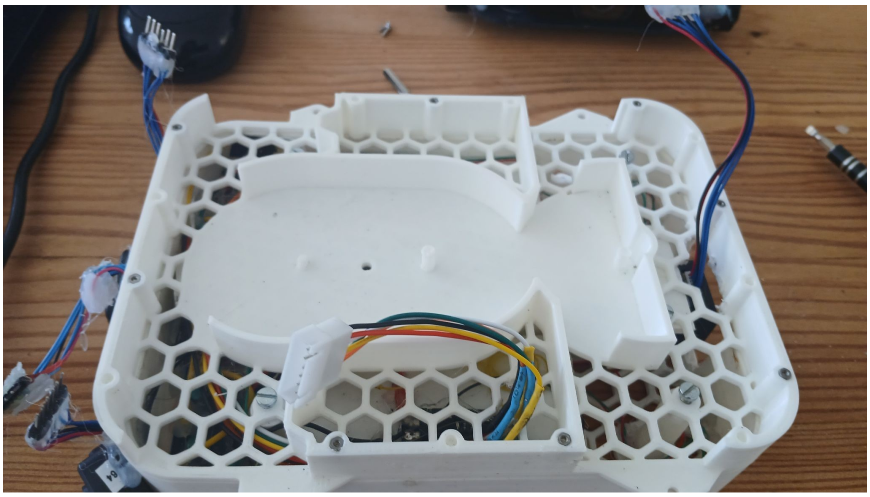


6) Assemble the rear steering mechanism. Place a 8mm M3 bolt into the countersunk hole on the gear, place this bolt through the steering rod. Tighten an M3 nut onto this bolt until you can easily push the steering rod with your finger, but it stops immediately after the pushing stops. Place a small amount of hot glue onto the nut. The images below outline the above process.

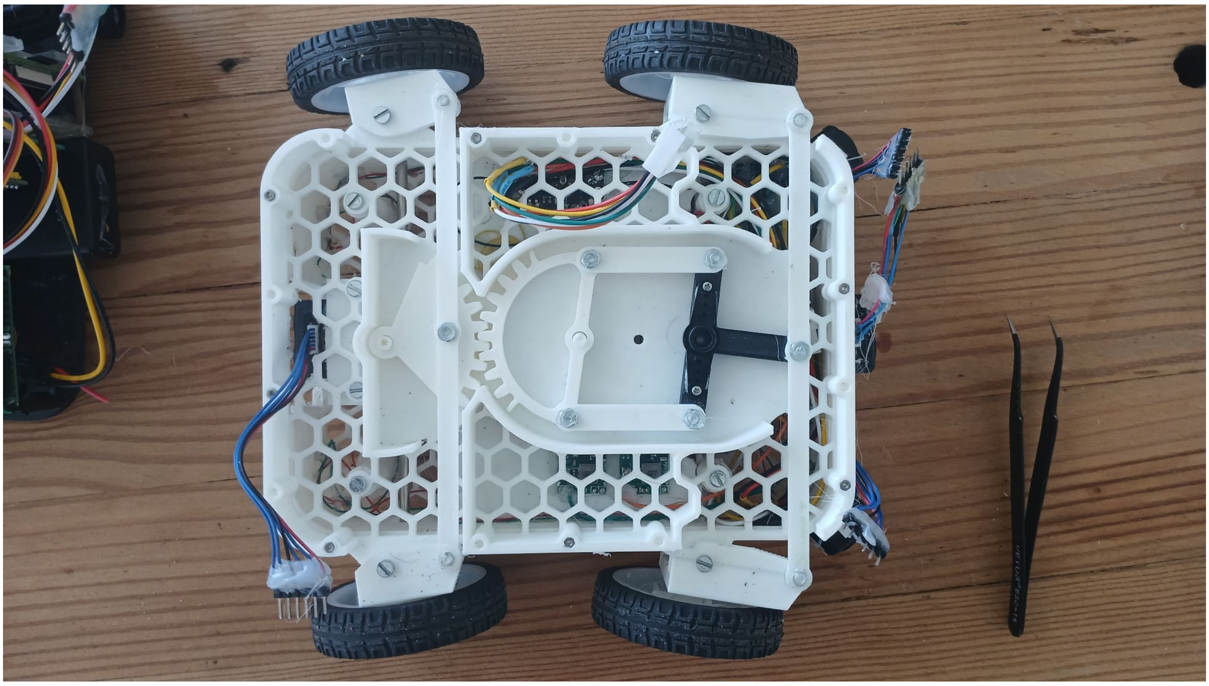


Final Assembly

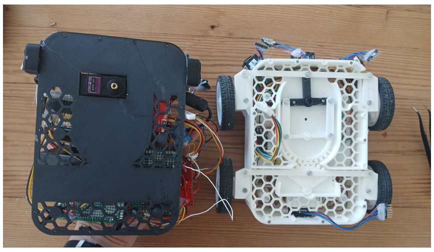
1) Attach the middle (main) frame and the lower frame. First, the pass-through connector needs to be connected between the middle and lower frames. Then, the frames need to be attached. This is achieved by aligning the rotational position of the motor and motor shaft. The main frame is then lowered onto the lower frame, with the gearbox mounted in-line with or touching the back wall of the lower frame. Once the main frame is resting on the lower frame, the main frame is pushed forwards, engaging the motor and motor shaft. Once this is complete the frame can be secured in place using 4, 6mm M3 bolts. The result of these actions should be represented by the image below.

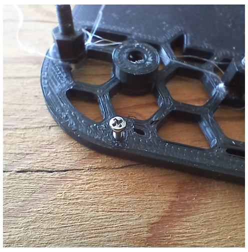


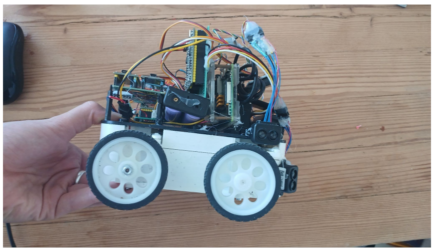
2) The wheels then need to be mounted to the combined frame. The main frame is placed between the two upper sections of the wheel mount and the lower section of the wheel mount rests on the lower frame. A 16mm M3 bolt is then inserted into the upper hole and secured with a lock nut. The same is repeated with the lower section, using a 12mm M3 bolt. These bolts should be tensioned similarly to those on the steering mechanism. When mounting the back wheels, ensure that the shafts engage the gearbox when mounting.
3) Place the steering mechanisms on the frame. The mounting points are sized so that there is only one position in which it will fit.
4) Insert 6mm M3 into the countersunk hole on the wheel mount. This should be screwed into its corresponding threaded insert on the steering mechanism. This should be repeated for all 4 wheel mounts. Place a small amount of hot glue on the bolt/threaded insert to prevent it from vibrating loose. Care needs to be taken in this step not to invert the vehicle while the steering mechanism is unsupported as it may fall out. The final product should look like the below image:



5) Mount the top frame to the rest of the vehicle. The wheels should be straightened and the servo centred prior to this operation. The top frame can then be carefully lowered onto the rest of the vehicle. Gently press down on the servo to engage it with the servo horn. This assembly can then be bolted down using the 8mm m1.25mm bolts. The process for this is outlined in the images below.



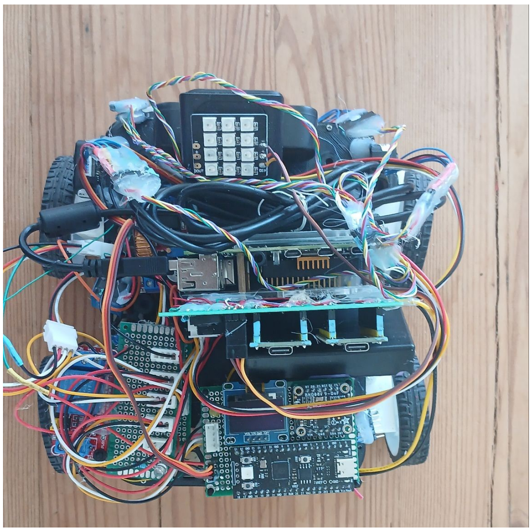


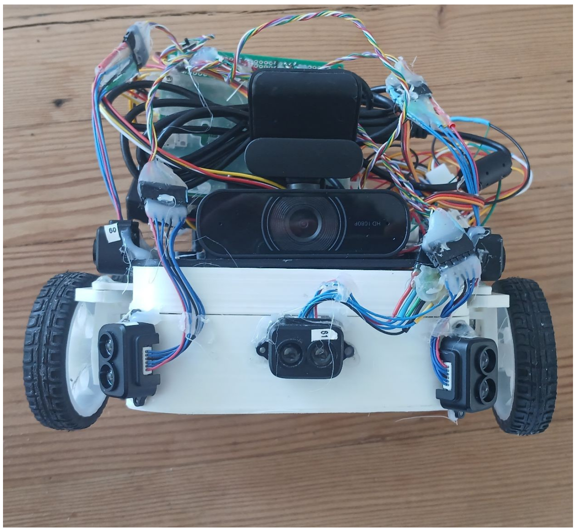


Photos of the Completed Vehicle

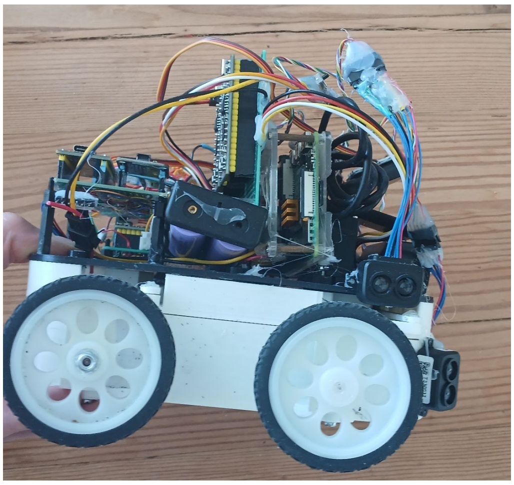
Additional (and higher resolution) images can be found here.

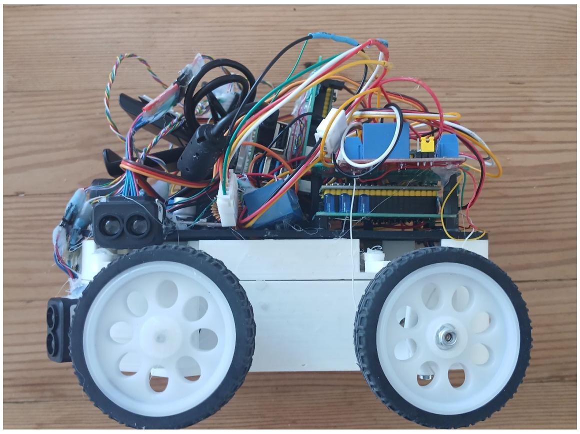
Top View **Front View**





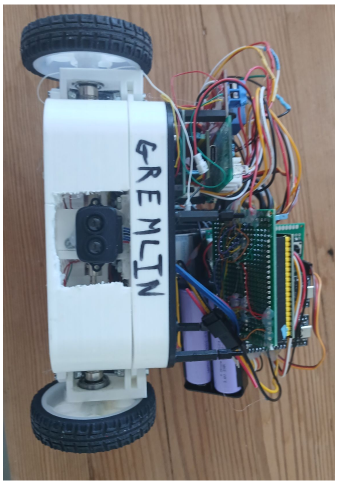
Side Views

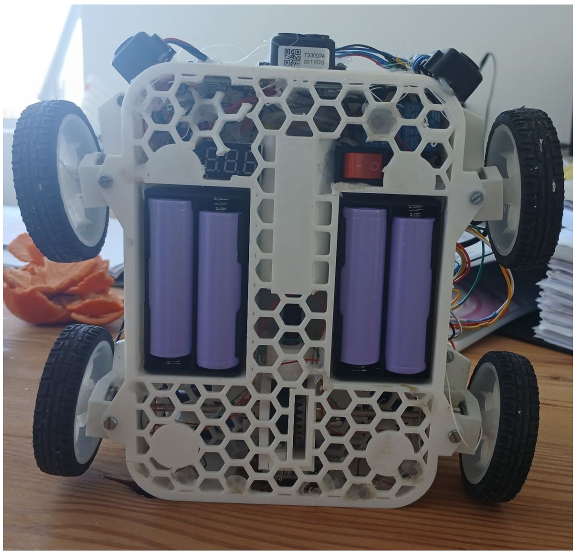




Back View

Bottom View





Pictures: Team

These images are available at a higher resolution here.

5. Performance Videos

Please see attached link for Team Null vehicle performance videos.

Open Category Clockwise:
https://drive.google.com/file/d/1u9WIaTdv1vVr_2TocBf5nXeMcmsZVSX0/view?usp=sharing

Open Category Anti-clockwise:
https://drive.google.com/file/d/1rUEM5G98ogTVsLS80-4uWdPW7KKapgZX/view?usp=sharin
g

Obstacle Course: Clockwise:
https://drive.google.com/file/d/1A3ZOPgexcJJU8Gloadm9QQgIXQ2pb4we/view?usp=sharing

Obstacle Course: Anti-clockwise:
https://drive.google.com/file/d/18CCVhjMQ0LsvGY3-RUxEsRtBEZ069Jtl/view?usp=sharing

Parking:
https://drive.google.com/file/d/1vMxz0yG7lwOoRhPndcRS1lcij8LYPWNm/view?usp=sharing

6. GitHub Utilisation

The Team Null Github for this project is available here.