Autonomous Driving Robot Project: Technical Report

ENGR122-C

Professor Anthony Russo

6 May 2022

Authors: Shannon Rhatigan, Jaime Bermudes, Mason Brewster

–

*"I pledge my Honor that I have abided by the Stevens Honor System"*

Signed: *Shannon Rhatigan, Jaime Bermudes, Mason Brewster*

**Table of Contents**

**Introduction & Requirements**

This project tasked engineers with creating an Autonomous Driving Robot for the presented client, Stevens Institute of Technology, that could guide visitors on campus tours of S.I.T. The product was expected to: reach four targets in a specific order, avoid external collisions, and collect location coordinate information from the MQTT network. The final task requires the installation of a GPS float, additional ultrasonic sensor(s), and an OLED display, so the robot body design needs to be updated to install all devices. The system was also expected to meet specific parameters that included: a pole diameter 0.31 inches, a robot height of less than 5 inches, at least 3 ultrasonic sensors, a mounted WeMos board, and no tape or glue to adhere portions of design together. The engineers were given a maximum time frame of six weeks to finalize all aspects of their system and completed the job with great success. The team was able to design a bracket to hold all of the required materials for a functioning robot, and the robot was able to hit 3 out of 4 targets, while avoiding all obstacles, throughout the arena. The purpose of this report was to allow for the team to break down various aspects of the project such as: the background, overall project objectives, system design, mechanical design, software requirements, wiring and more.

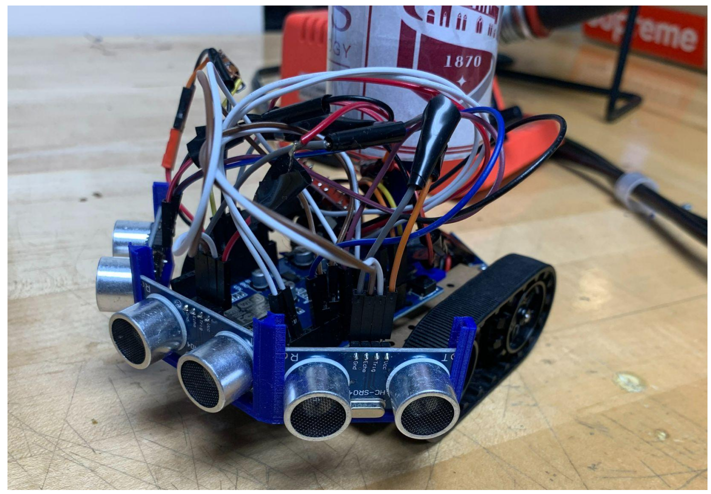
[Author: Rhatigan]

**Concept Design**

The overall approach to the design concept was to make a simple, minimal, easy-to-print bracket design, so that the team could achieve the 3D printing time constraint of 4 hours or less. The design had to be lightweight enough to avoid weignhing down the robot by being disproportionately heavy, while also being sturdy enough to support the weight of the provided sensors, motors, and wiring system. The first designs that had been set to print were failing, as the base of the bracket was having trouble adhering to the build plate of the printer.

**Final Design Description**

For the group's final design, the bracket took 2 hours and 40 minutes to complete printing. It was able to hold the WeMos Board, the GPS float, and the Ultrasonic Sensors successfully, without imbalacnce of the overall structure, as we intentionally designed the bracket with consideration for the center of gravity and component placement. The final bill of materials utilized included the following:



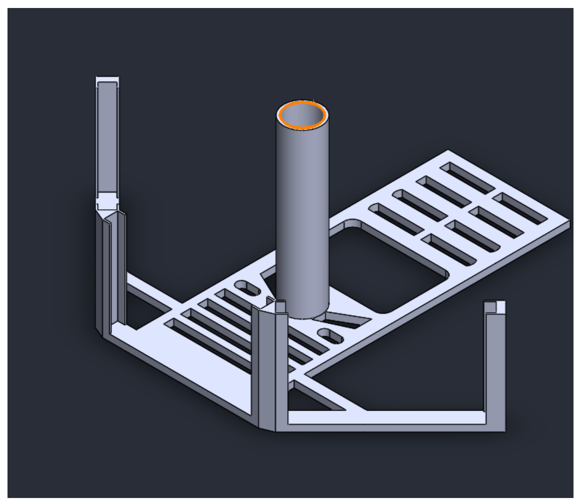
- 3D printed platform
- WeMos Board
- LiDAR sensor
- GPS Float
- Motors
- Robot Chassis
- Battery
- Ultrasonic Distance Sensor
- OLED Display
- Wires

*System Description and Mechanical Design*

The robot was designed to meet the requirements set in the project briefing. The robot made use of three ultrasonic sensors to determine distance from obstacles and walls, made use of a WiFi enabled WeMos board to retrieve information from a LiDAR system to determine position of the robot, and used an OLED display to display all of this information. To use all of these parts simultaneously a platform needed to be designed to hold each part of the robot. The mount for the three ultrasonic sensors was designed to be in the front of the robot so that it could avoid obstacles and not run into walls. The platform was designed to hold the WeMos board in the middle of the robot to maintain balance and ensure ease of use and maintenance.

For the LiDAR system to work properly, a gps float was needed. The mount for the gps float was positioned in the rear of the platform to act as a counter balance for the entire robot to minimize “jerking” or tipping too far to the front. The mount for the OLED display was designed to be on the rear of the mount for the gps float so that the user could easily view the information it displayed without having to peer through wires or interrupt the obstacle avoidance system.

All of this hardware had to fit on a robot chassis provided to the group. For this reason, the platform had a modular design. The modular design has what is called a “channel” system, meaning there are no screw holes, just channels that can fit the screws to allow for any errors in the measurements of the robot chassis. This modular design allowed the arduino board to be mounted anywhere on the platform without interfering with the mounting of the platform on the robot chassis. This modular design, due to its “channel” system also minimized the weight of the platform as well as the time it took to 3D print.



Multiple interactions of the platform were designed, printed and tested. Early prints of the platform failed due to the 3D printer, not the platform itself. The version used on the final robot was the fourth iteration of the design. This final iteration met all of the requirements presented in the project briefing and worked well.

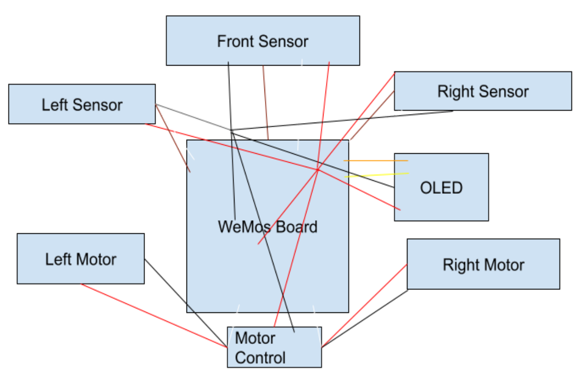
*Electrical and Wiring Design*

To ensure that all of the components worked and functioned as they should, they had to be wired properly. The source of power for all of the components was a battery pack. The electricity was routed out the battery pack, through the WeMos board and into each component of the robot. One of the requirements for the wiring was that a breadboard was not allowed to be used. To accommodate this requirement, ground and voltage wires for each component needed to be soldered together such that each ground and voltage wire had four outputs to provide electricity to the ultrasonic sensors and the OLED display, the motors were provided electricity with their own dedicated ground and voltage wires.

The WeMos board is responsible for sending and receiving signals to and from each component of the robot as well as interpreting the received signals into directions for the robot. For each ultrasonic sensor a trigger and echo wire needed to be assigned an individual output port on the WeMos board. The trigger is the wire in which the WeMos would send a signal to start the ultrasonic sensor, the echo is the wire in which the WeMos would receive this feedback signal from the ultrasonic sensor. The trigger wires were brown and the echo wires were white so they could be easily interchanged. The OLED also required two individual ports on the WeMos board so that information the WeMos board collected from the LiDAR and ultrasonic sensors, as well as the interpreted information, could be displayed. The SCL wire was colored yellow and the SDA wire was colored orange.

The motors were reliant on a motor controller that received power from the WeMos board on a dedicated voltage and ground line. The motor control board required two dedicated signal wires to be plugged into the WeMos board so that each motor could be controlled individually using the information the WeMos board receives. These wires were color coded as white.

The majority of the wires were also cut and soldered down to fit the chassis and platform better so that loose wires would not interfere with the LiDAR system and alter the information the WeMos board receives from it.




*Software Design and Coding*

The code was built using three steps. Inputs, analysis, and outputs. The code has the framework to take in and store the information from the available sensors. Then it uses an algorithm to analyze the sensor inputs. This then commands our outputs.

First defined were the inputs and outputs utilized. The board received inputs of its x and y coordinates from the lidar system through a cloud subscription model. It received distance measurements from each of the three sensors. The only outputs used were for the left and right motors.

To collect the location data from the server the WeMos board was first connected to the wifi network using the provided id and password. The wemos waits until it is connected to wifi to do anything.  It subscribed to mqtt server address, and then subscribed to a topic. To collect data from the sensors, the board first stores the pin locations of the sensors. The sensors are then defined as an object which can be called upon using a command.

To control the motors, there was a defined object for each of their motors and associated pins. This allows the wemos to set the motor speed using a simple command. The motor speed controlling system inputs 0 as the fastest speed, 90 as stop, 180 as the fastest speed in reverse. Because this can be confusing there was functions to do common maneuvers including, forward, reverse, stopping, turning left, and turning right. The motors could also be controlled separate from the function by directly setting the motor speed, something we would take advantage of.

The analysis portion of the code was broken down into three categories, pathfinding, target hit, and obstacle avoidance. The obstacle avoidance was there to avoid hitting any of the walls, while pathfinding helped the robot work towards the target. The pathfinding code was continuously run, but would be overridden if the obstacle avoidance code activated. Target acquisition code took precedence over both.

The target acquisition code used the latest x and y coordinates to calculate the distance to the target. Target locations were stored using an array, so that it could easily be switched to the next target. If the distance calculated was under 150, then our robot would stop at the target. Depending on the target acquired it would either make a 90 or 180 degree turn to point in the direction of the next target. We changed the selection of the array to the next target once one was acquired.

The pathfinding algorithm used the previous location and the current location to determine a vector for the direction the robot was going. The previous locations were stored in an array allowing us to use points further away from the current location to increase accuracy. We used the newest location that provided a vector of approximately 25 units. If none of the recent points were over 25 units, the algorithm used the furthest back point by default. It then calculated a vector for the direction the robot needs to move using the most recent position and the position of the current target. The angle needed to be turned could be calculated using the dot product and magnitudes of the two vectors. The direction the robot needs to travel was determined using the cross product. A positive cross produced indicates turning left, while negative determines right. Based upon the direction, the left or right motor would be set to a speed higher than the other motor. The difference was based upon the magnitude of the angle. This allowed the turning radius to be higher or lower depending on how tightly the robot must turn.

Obstacle avoidance relied on the sensor inputs. It first updated the distance from all of the sensors. If an obstacle was directly in front of the robot a 90 degree turn would be made. The direction it would turn was first determined by whether or not there was an additional obstacle on either side, then by the direction determined by the pathfinding algorithm. If directly ahead was clear, then the robot just slowly turned away from the wall or obstacle. This was done by increasing the speed of the motor on the wall side by a function of how close the robot was to that side. Any obstacle avoidance would override the pathfinding algorithm and restart the robot going straight ahead.

**Assembly and Prototyping**

To assemble the system, the 3D printed bracket that the engineers had designed, was mounted onto the provided robot chassis. At the base of the bracket was the WeMos Board, to the front-middle, front-left, and front-right, were sensors. Also attached to the system were the left and right motors, the motor control, as well as the OLED. As the team experimented with various bracket prototypes, the code was adjusted to provide for the most accurate and precise robot feasible.

**Performance Reporting**

Overall, the group's design was largely successful. The system was great at cautiously maneuvering around the arena, with the ability to squeeze through tight spaces in between obstacles, and never hitting the wall. However, the robot was only able to hit ¾ of the intended targets, as issues arose once the robot began its journey from the 3rd target towards the last, where it would essentially oscilate around the targets, without ever hitting the final one.

**Reccomendations and Conclusions**

The group concluded that they were able to successfully manufacture a robot, proficient at avoiding walls and obstacles around a potential campus like that of the client. However, the engineers struggled with time, as the first few attempts at printing the 3D bracket failed, providing a minor setback, as well as complications with hitting all 4 presented location targets. Advice that the team would give to future engineers to aid in their victory with the completion of this project would be to clearly outline your process, create a solid design plan, and maintain good inter-team communication, throughout the semester.