Grand Theft Autonomous:
MEAM 5100 Final Project

Sanghyub Lee
sanghyub@seas.upenn.edu

Pranav Gunreddy
grpranav@seas.upenn.edu

Renu Reddy Kasala
renu1@seas.upenn.edu

I. Overview

This project entails the development of a mobile autonomous robot with advanced sensing and communication capabilities. The robot exhibits autonomous wall-following behavior, precise localization through Vive system integration, and employs infrared beacon tracking for targeted navigation. Notably, it communicates using communication protocols such as UDP and ESPNow, enabling dynamic interaction. The robot's proficiency is demonstrated by its ability to autonomously manipulate objects, including grabbing a real trophy, pushing a fake trophy and a police car with precision, highlighting its adaptability in practical applications

II. Approach

While our primary focus was on achieving specific functionalities for the robot, during the brainstorming and design phase, we adopted a conventional approach. This involved structuring our robot pipeline to encompass key components such as perception, localization, planning and control, as well as system integration and communications. By prioritizing these fundamental behaviors and software/hardware aspects, we aimed to ensure a robust design and effective control for our autonomous mobile robot.

A. Sensing/Perception

In the sensor domain of our robot, we integrated infrared sensors, employing carefully crafted detection circuits that utilized opamps and comparators to eliminate noise and precisely detect the frequency of emitted IR light. Simultaneously, ultrasonic sensors were incorporated with software filters to refine readings, enhancing accuracy. These sensor outputs were seamlessly integrated into our autonomous maneuvering strategies, forming a robust sensing foundation for the robot's navigation and task-specific functionalities.

B. Localization

We employed HTC Vive which is a GPS-like localization system to precisely determine the robot's position within its environment. To achieve this, we designed and implemented detection circuits that interfaced with the Vive's tracking sensors, allowing the robot to gather real-time data on its spatial coordinates. We performed calibration to ensure accurate alignment between the Vive system and the robot's internal coordinate system, enabling reliable and consistent localization

To determine the robot's orientation, a magnetometer was utilized to obtain relative orientation information. Integrating this data with the HTC Vive's position coordinates allowed us to establish a comprehensive localization system for the autonomous mobile robot.

C. Planning and obstacle avoidance

In our planning and obstacle avoidance strategy, we leverage the start position obtained from Vive localization and the end goal defined by either IR frequency recognition or Vive coordinates. Employing a simplified path planning approach, we prioritize ease of control over optimality. Utilizing sensor inputs from infrared and ultrasonic sensors, we implement reactive obstacle avoidance mechanisms to navigate around potential obstacles such as other vehicles and beacons. This approach ensures the autonomous mobile robot efficiently maneuvers through the designated field, adhering to the defined path while dynamically adapting to its surroundings.

D. Control

In our control strategy, we implement PID controllers for both lateral (angle/orientation) and longitudinal (distance) control. The desired distance and orientation, dictated by the planning module, guide the robot's movements. In instances requiring sudden obstacle avoidance, rapid turns are executed based on real-time input from ultrasonic sensors, ensuring swift and effective maneuvering in response to dynamic environmental changes.

E. Web Interface design and communications

For interface, we hosted a HTML based webpage and employed UDP WiFi communication, enabling interaction with the robot. The webpage featured task selection options, a stop button, and flags indicating task completion or timeout. Minimizing packet transmission was crucial for promoting autonomous behavior and earning bonus points. Additionally, we implemented ESPNow communication for seamless message transfer between the robot's ESP microcontroller and the pilot microcontroller, enhancing the efficiency of command delivery from the webpage to the robot.

The robot herein was required to demonstrate three primary functionalities, as mandated by a competition scenario.

- The robot showcases autonomous wall-following behavior
- It adeptly tracks Infrared emitters emitting frequencies of 23 Hz and 550 Hz

- It exhibits autonomous mobility by navigating to a randomly positioned police car box

III. Mechanical Design

The mechanical design of the robot facilitates cost efficiency while providing ease in maneuvering, navigation through obstacles/ collisions and in place rotation. we have designed a differential drive robot with two wheels at the centre and two castor wheels in the front and back. The design focuses on minute details about chassis design, sensor and motor placement, wire management and a single gripper mechanism for pushing police cars and grabbing beacons without additional components. Here's an overview of the key considerations:

1) Chassis: Robot's chassis has been made with 0.125" acrylic boards which were lasercut to accomodate all mechanical and electrical components. The three-layer acrylic chassis provides a sturdy and modular framework while keeping the bot very light. we also ensured we stayed under dimension constraints of 12"x12"x12". we used stand offs and appropriate fasteners for easy access to all parts which would make upgrades and maintenance smoother.

The layered structure segregates components into three layers.

- Layer 1 includes motors, IR sensors, ultrasonic sensor, gripper servos, power bank, and battery, with appropriately sized holes for safe wire routing.
- Layer 2 hosts the ESP 32S, motor driver module, and magnetometer module.
- Layer 3 contains the Vive circuit and IR detection circuits.

2) Motors: The 12V DC geared motors with a speed of 1000 RPM and great power torque specs have been chosen and strategically placed to provide maximum traction. The flat rubber tires enhance contact area with the surface, ensuring better grip and maneuverability. Threaded tyres usually lost balance or slipped. The placement of motors contributes to the overall stability and balance of the robot inclusing factors such as centre of mass and ground clearance since they were the heaviest parts of the assembly.

3) Sensor Mounting: To eliminate noise and enhance sensing accuracy, mounts for IR, Vive and ultrasonic sensors are incorporated. 2 IR sensors have been strategically placed at optimal distance and height matching that of the beacon to be able to effectively gauge the strength of infrared signal from the source and position itself accordingly but also improve the probability of sensing without extensive search. The 3D printed minimal design of IR mounts is tailored for efficient signal reception. Ultrasonic and magnetometer sensors are placed to provide accurate data for navigation and orientation. Ultrasonic sensors position, specifically needed lots of tuning to result in the right distance from the walls and not from ground or garbage values

4) Gripper Mechanism: The single gripper mechanism is designed for dual functionality – pushing police cars and grabbing beacons. Careful consideration is given to positioning the gripper below the center of mass of the beacon to prevent toppling during grabbing. Tight grab of beacons has also been made possible so that beacon pull is also effective and not just push. when fully closed, the grippers provide a sturdy base to effectively push the police car. keeping the police car's weight in mind, we added supports and maximized contact for uniform pushing. The gripper is activated by servos, allowing precise and controlled movements.

5) Wire Management: Appropriate holes in the chassis are designed for safe routing of wires, preventing tangling and interference with moving parts. This ensures the longevity and reliability of the robot by minimizing the risk of damage to sensitive components.

IV. Electrical Design

The focus on the electrical design was simplicity. Electrical management and maintenance are crucial to active testing and evaluations. Therefore, the electrical design needed to be compact and easy to maintain.

1) Microcontroller: The ESP 32 S2 SAOLA was selected as a single controller for the robot. The goal was to only use a single microcontroller to make the circuit design simpler. Therefore, S2 SAOLA was selected because of its numerous pins with many ADC pins allowing the robot to operate on a single microcontroller.

2) Motor Circuit: The L298 Dual H bridge, DC motor driver powered by Lipo battery (1500mAh, 7.5V), was used to control two DC 12V gear motors for differential drive. The L298N has huge advantages in current capability and the dual bridges allow the separate control of two DC motors. The motor has a built-in encoder allowing PID control to be easily achieved. The two cell lipo battery has enough current and voltage to power the motor driver, and the lipo battery has higher energy density than other batteries allowing the robot to be lighter and still powerful to drive.

3) Sensor Circuit: There are several sensor circuits used for the robot: Vive sensor (PD-70) circuit, IR sensor(LTR-4206) circuit, and Ultrasound sensor(HC-SR04) circuit. As shown in the figure below, TLV272 op-amp was used for both the Vive circuit and IR sensor circuit. TLV272 op-amps are relatively immune to noise and are suitable for precise signal amplification. Therefore, it was selected to be used in a sensor circuit that is either detecting or receiving a signal. For the IR sensor circuit, the feedback resistors were carefully tuned to be able to detect and differentiate signals up to 2.5 meters. The ultrasound sensor had also built-in trig and echo

pins allowing a direct connection with the microcontroller.

4) Design process: Each circuit was separately tested on the breadboard before soldering the components to ensure functionality and simplicity. The pins on the microcontroller were selected according to the proximity of the element to the pins. Then, all the seperate circuit was integrated into a working electrical system for the operation of the robot.

V. Process Architecture

We used a single microcontroller ESP32 S2 in our configuration, as illustrated in below. When interacted with the HTML webpage, user inputs are transmitted over Wifi via UDP to the microcontroller. Our computer and ESP32 S2 device are equipped to send, receive and process various broadcasted UDP messages. These messages include Robot identity, Vive location of our robot and Vive location of the police car. Messages are also broadcasted at the end of each task ensuring the user can initiate the next task without being in the vicinity of the playing field.

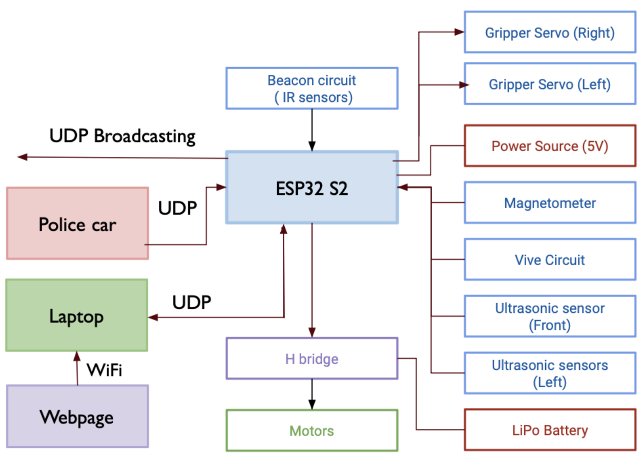

Fig. 1. System high level architecture

VI. Software Architecture

A. Wall Following

Initially, we implemented PID control with motor encoders to ensure straight-line navigation. Subsequently, we utilized ultrasonic sensors located at the front and right sides. Employing PID control, the robot maintained a safe distance from the right wall while moving straight until reaching a predefined threshold. Upon reaching this threshold, a lateral controller was activated to execute a precise 90-degree turn. To verify the correct orientation post-turn, ultrasonic readings were again employed. Through repeated experiments, we found out appropriate speeds to travel at which would ensure we would sail smoothly on a rather bumpy, uneven surface but also finish tasks fast to score maximum points

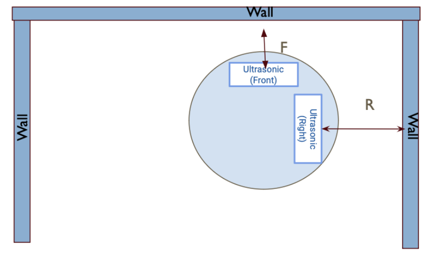

Fig. 2. Wall following mode

Algorithm 1: Wall Following Behavior

Data: Parameters d

Result: Wall following behavior

`reachWall` ();
repeat
  `moveForward` ();
  if $(F > d)$ **then**
    if $(R < d)$ **then**
      `turnLeft` $(R \approx d)$;
    end
    else
      `turnRight` $(R \approx d)$;
    end
  end
  if $(F < d)$ **then**
    if $(R < d)$ **then**
      `stop` ();
      `turnLeft` $(R \approx d)$;
    end
  end
until *until stopped or task accomplished*;

B. Beacon Tracking

For beacon tracking, our robot is equipped with two strategically positioned IR sensors capable of detecting frequencies at 23 Hz and 550 Hz. The algorithm orchestrates precise navigation towards the beacon, ensuring a head-on orientation while maintaining stability to prevent toppling. To interact with the beacon, servo motors are employed to actuate our gripping mechanism

C. Police Car Pushing

Utilizing the initial Vive coordinates of both the robot and the police car, our algorithm orchestrates precise navigation to the center of the police car, ensuring correct orientation for the closed gripper to face the car's center. To account for potential inaccuracies in Vive data, ultrasonic sensor information is also fused for localization. Upon reaching a designated threshold, the robot halts so that it can cruise at max speed with sufficient momentum to actively engaging in pushing the police car. The algorithm iteratively verifies the displacement using the police

Algorithm 2: Beacon Tracking Behavior

Data: IR sensor states IR, Frequency comparison function getFreq(IR), Parameters threshold δ
Result: Beacon tracking behavior

```
FindBeacon ()
BeaconTracking ()
if (550 − δ < getFreq(IR) < 550 + δ) then
    grabTrophy ();
end
else if (23 − δ < getFreq(IR) < 23 + δ) then
    pushFakeTrophy ();
end
grabTrophy ()
moveForward ();
if (F < d) then
    gripperGrabMode ();
    Position ← getCurrentPosition ();
    moveTo (predeterminedPositionOurSide);
    gripperOpenMode ();
end
pushFakeTrophy ()
moveForward ();
if (F < d) then
    gripperGrabMode ();
    Position ← getCurrentPosition ();
    moveTo (predeterminedPositionOpponentSide);
    gripperOpenMode ();
end
```

car's Vive coordinates every second, ensuring the car is pushed beyond a specified distance of 10 inches.

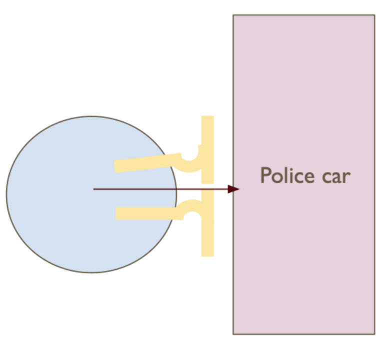

Fig. 3. Push police car mode

VII. Performance

We successfully achieved all functionalities required by the robot and checked off with extra credit - Control the robot, Wall Following autonomously, Autonomously identify, go to and push either trophy or fake (using 23Hz or 550Hz beacon tracking), Use the Vive system to transmit your X-Y location,

Algorithm 3: Police Car Pushing Behavior

Data: Vive positional data, magnetometer data (angle), Parameters d, δ, θ, α
Result: Police car pushing behavior

```
getCurrentPosition () From Vive;
getPoliceCarPosition () From Vive;
pushPoliceCar ()
initialPosition ← getCurrentPosition ();
policeCarPosition ← getPoliceCarPosition ();
d ← calculateDistance (initialPosition,
  policeCarPosition);
α ← getAngle (); From magnetometer
θ ← (initialPosition, policeCarPosition);
rotate(turnAngleCalc(θ, α));
moveToCoordinate (policeCarPosition);
if d ≤ threshold then
    while (movement ¡ 10) do
        gripperPushMode ();
        moveForward(10 inches);
    end
    finalPoliceCarPosition ←
      getPoliceCarPosition ();
    movement ← calculateDistance
      (policeCarPosition, finalPoliceCarPosition);
end
```

Autonomously move to the police car box and push it 12 inches.

We unfortunately couldn't pariticipate in the competition due to engagements in other courses and unreliability of vive sensing but approached the project, made the bot keeping the competition in mind

A. Challenges & lessons learnt

Engaging in such a comprehensive project involving entire lifecycle of a mobile bot, comes with its fair share of challenges. some of them are silliest of mistakes that kept us up multiple nights but have been a major source of learning for us and complimented concepts taught in the class

- Achieving precise CAD design with accurate dimensioning and attention to detail to ensure components fit together seamlessly. But this made sure we didn't have multiple manufacturing iterations and waste material
- Component Integation has been a major source of error and ambiguity and hence had to be dealt with utmost precision. Use of correct jumper wires that had a strong fit was important since connections got loose on impact/ collision. wire rerouting and other noise reduction techniques taught in class came handy
- Ultrasonic sensor mounting is an example of how key mounting is. Since the diameter of the bot is pretty large, our initial idea of placing ultrasonic sensors on diametric ends didn't work for us.

- Power electronics is pretty dicey. We ended up blowing microcontrollers, motor drivers, op amps due to irregular handling of power connections. Meticulous testing of soldered components is also important due to risk of shorting. A multimeter should be carried around every time to check if connections are in place
- Print statements cause a lot of delay that isn't insignificant. we lost sensor information (or it came in late) at crucial times due to lots of print statements. For example cramming into wall during wall following when its supposed to stop/ turn.
- Sensor Calibration: The sensor calibration was an arduous process of finding an optimal sensor reading for different tasks to be executed. Also, finding an optimal range of resistors to use to minimize the noise and get clear readings from the sensors. To allow for faster and more accurate calibrations, all sensor calibration was conducted on the test track with an oscilloscope to actively monitor the sensor readings and adjust according to the track environment.
- Damage control: The PD70 sensor in the vive circuit had to be replaced several times because of its fragility. Also, resolving issues in soldered parts was a challenge because all the elements are connected limiting the change or active testing on the circuit. To address these challenges and to solve the issue, Digital Multimeter connectivity tests were thoroughly conducted to check for connectivity issues and damages. Also, adding casing to the circuit really minimizes the damage during evaluation and testing

VIII. Appendix

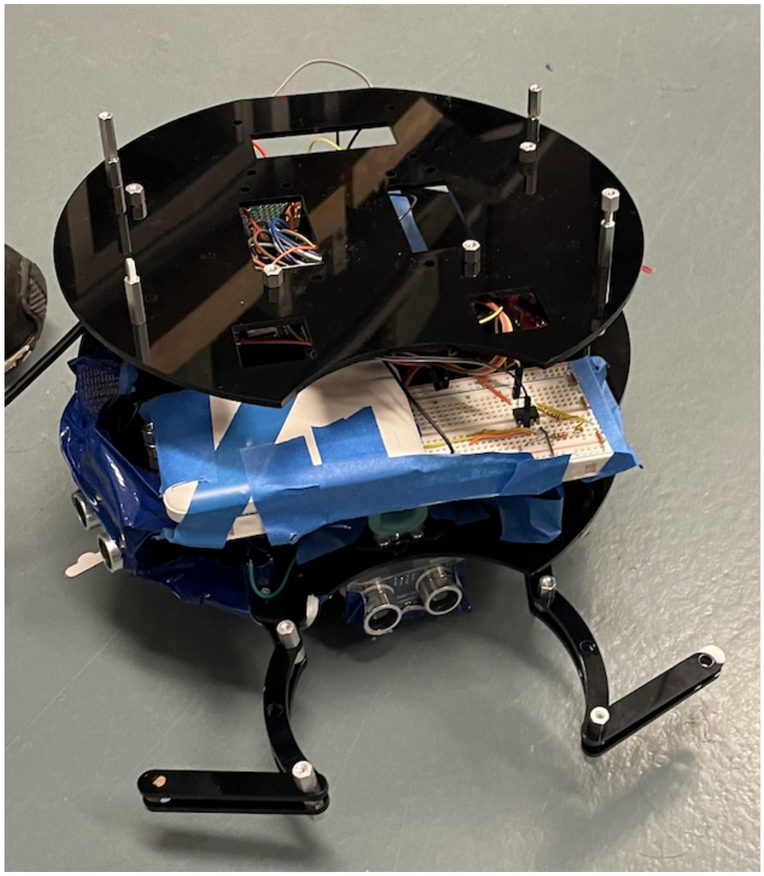

Fig. 4. Final Assembled bot

Bill of Materials:

#	Item	Link	Quantity	Price
1	DC Motors with encoders, coupling and wheel	Link	2	$43.0
2	Roller Ball Bearing	Link	2	$4.98
3	Lithium polymer battery pack	From TA	1	$7.50
4	Miady USB charger	From TA	1	$7.00
5	Ultrasonic range sensors	Link	5	$8.00
6	Magnetometer	Link	1	$7.00
7	L298 DC Dual H Bridge Motor Speed Controller	Link	1	$12.0
8	Perfboarb, Connectors, Wago splices, header pins, acrylic board, Op amps etc..	Ministore	-	$0.0
			Total:	$89.48

CAD Drawings:

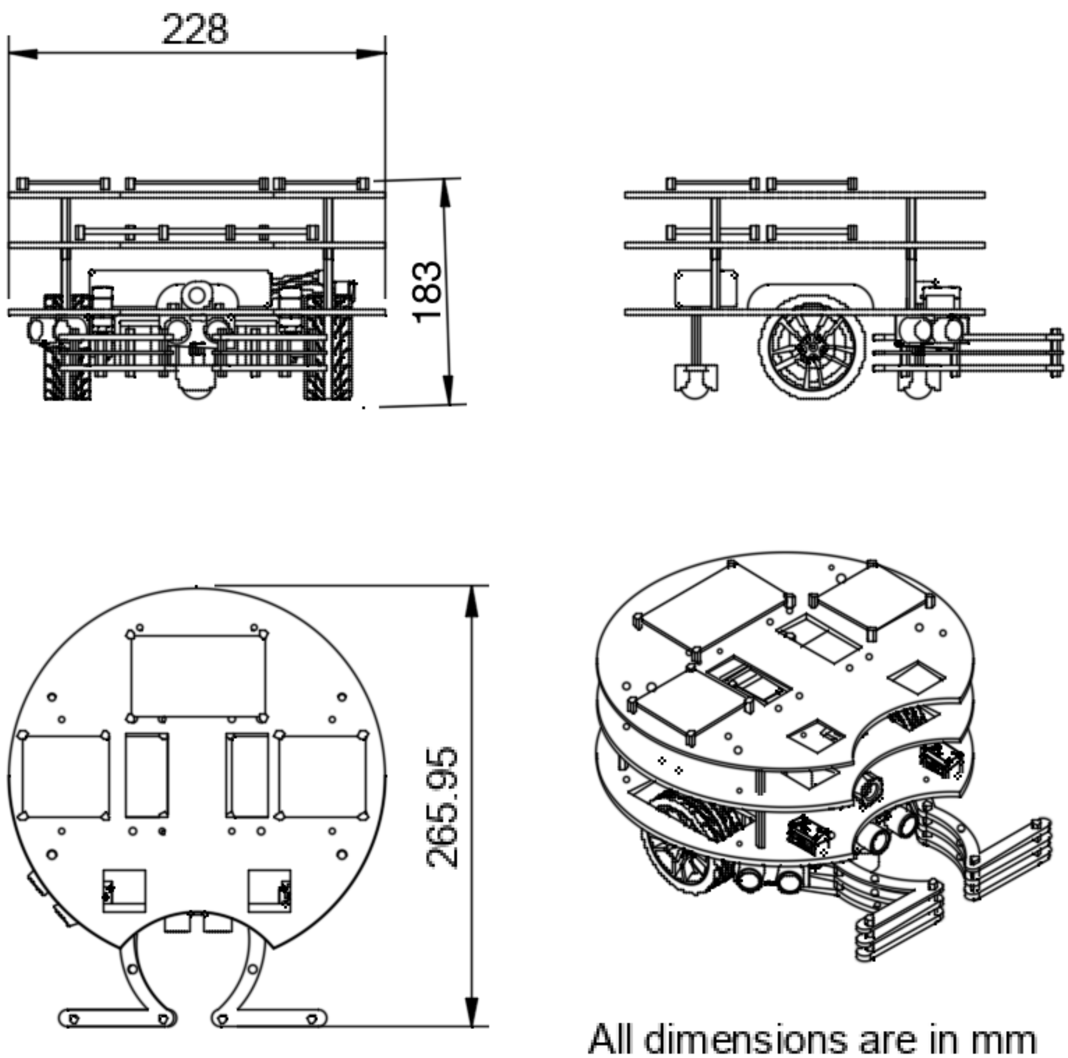

Fig. 5. 2D Drawing

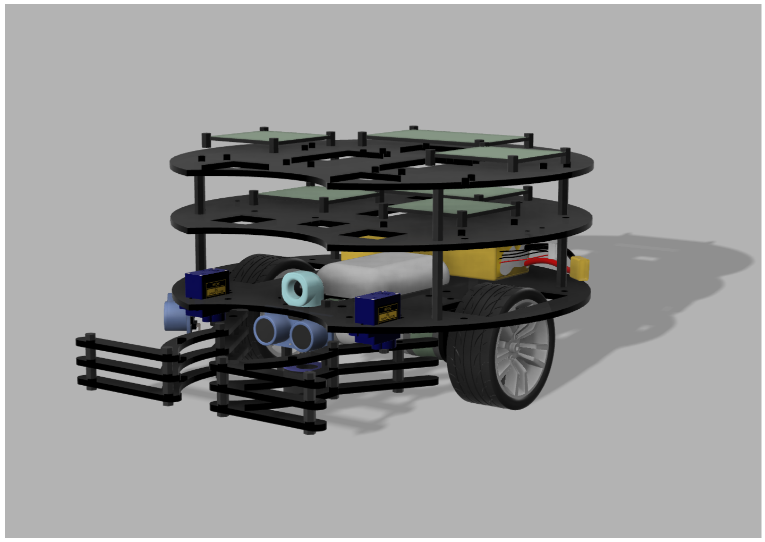

Fig. 6. CAD: Isometric view

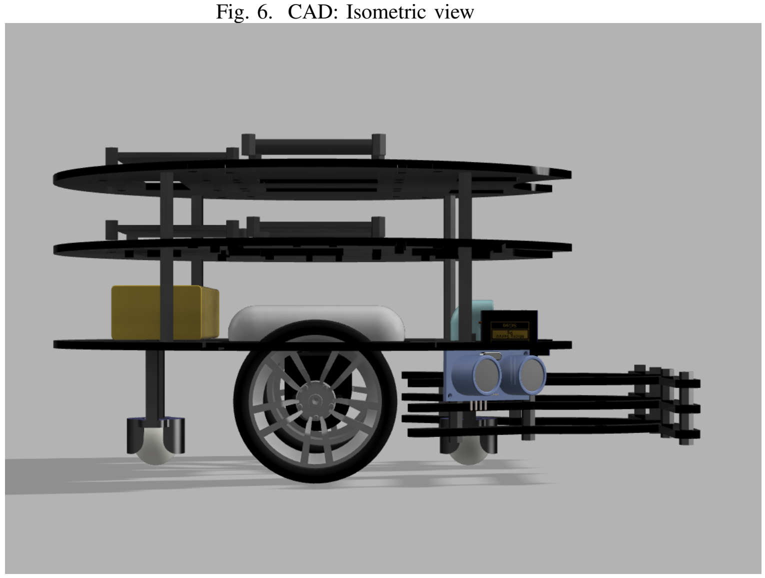

Fig. 7. CAD: Right side view

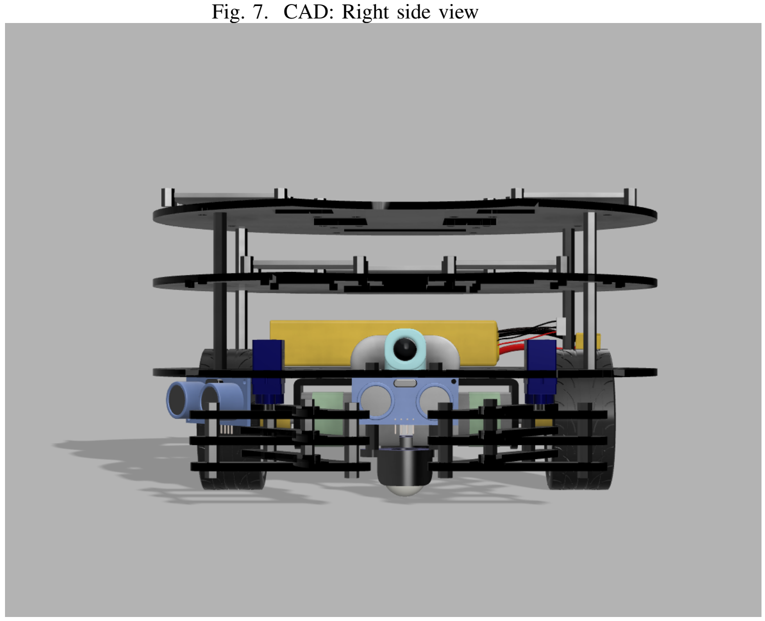

Fig. 8. CAD: Front view

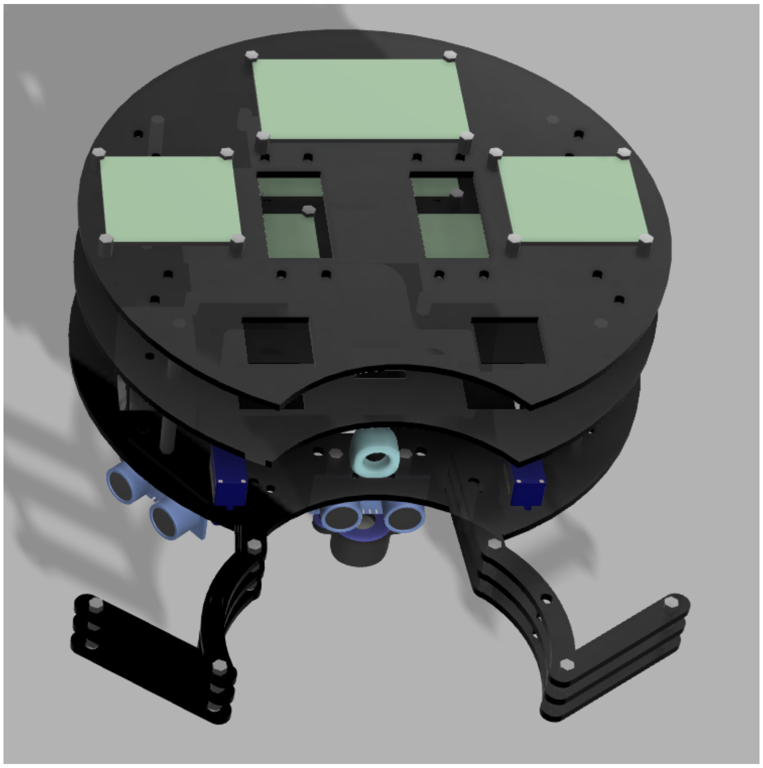

Fig. 9. CAD: Beacon grabbing mode gripper positioning top angle view

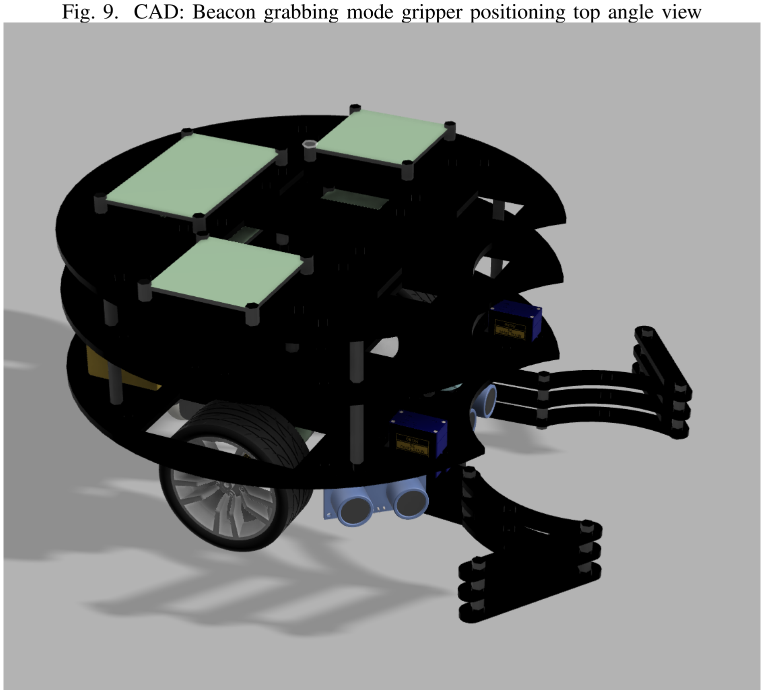

Fig. 10. CAD: Beacon grabbing mode gripper positioning

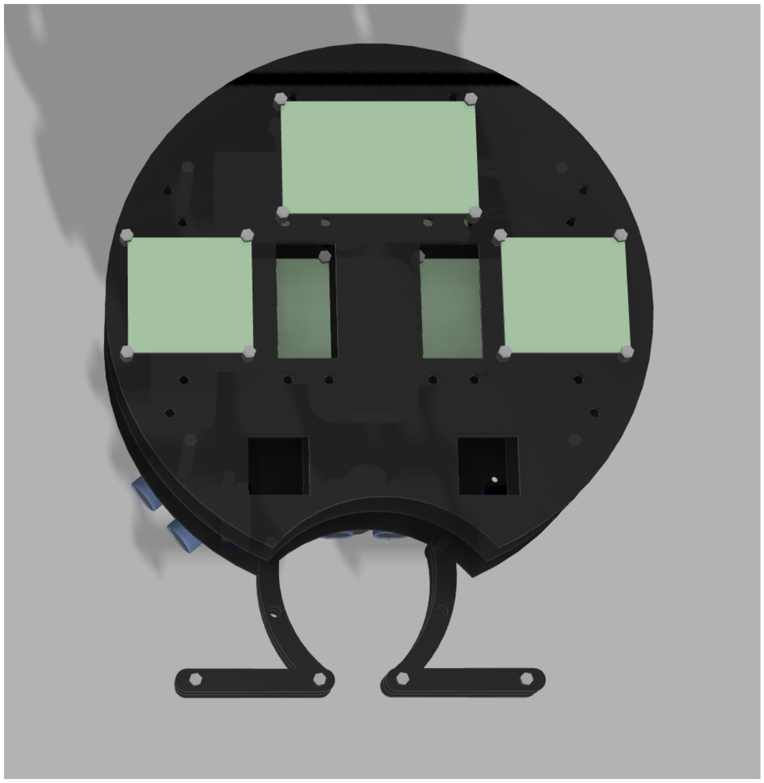

Fig. 11. CAD: Push mode gripper positioning top angle view

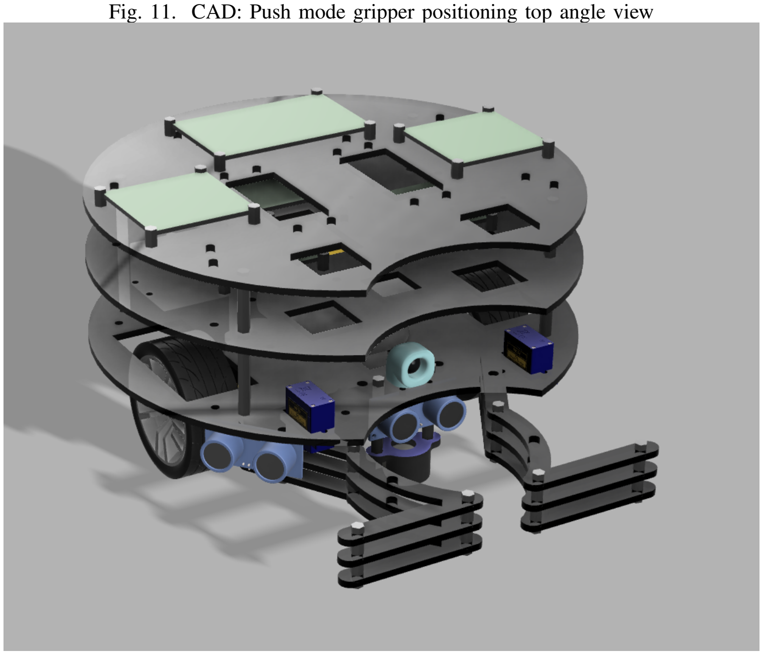

Fig. 12. CAD: Push mode gripper positioning

Layer wise sensor positioning

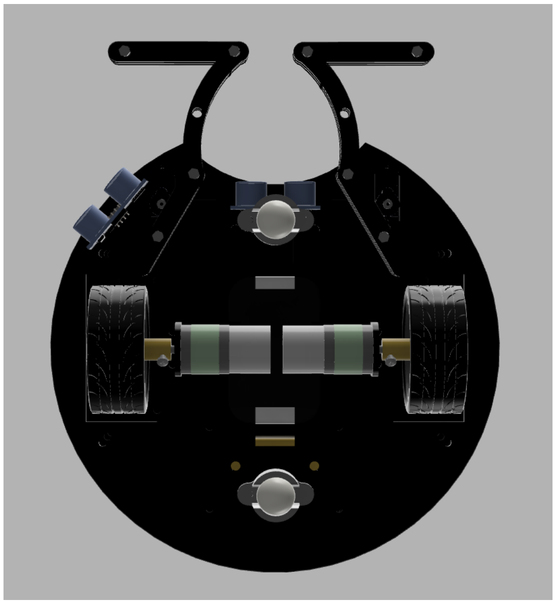

Fig. 13. Layer 1: Bottom side

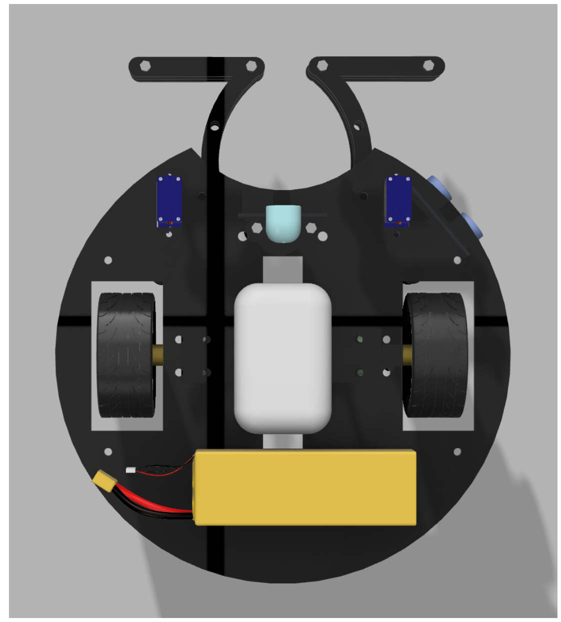

Fig. 14. Layer 1: Top side

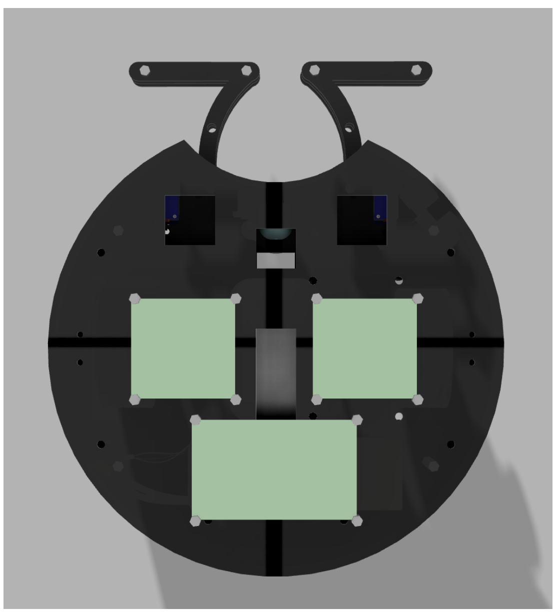

Fig. 15. Layer 2: Top side

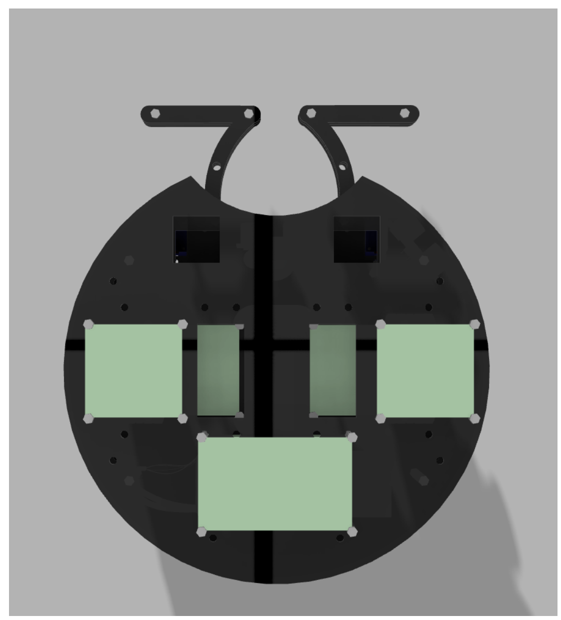

Fig. 16. Layer 3: Top side

Zoom in view for mounts:

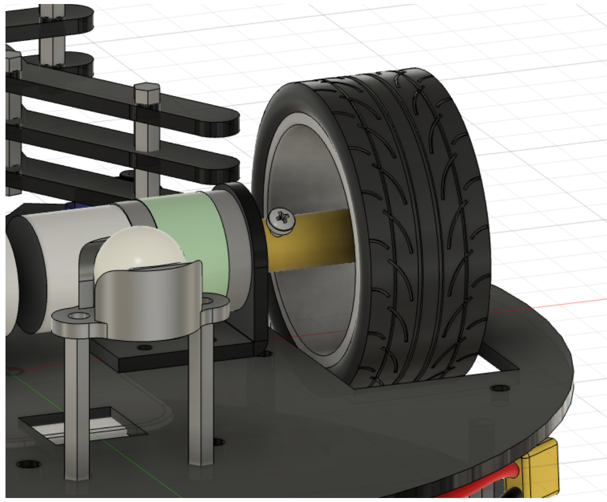

Fig. 17. Motor mounting via Hub and coupling

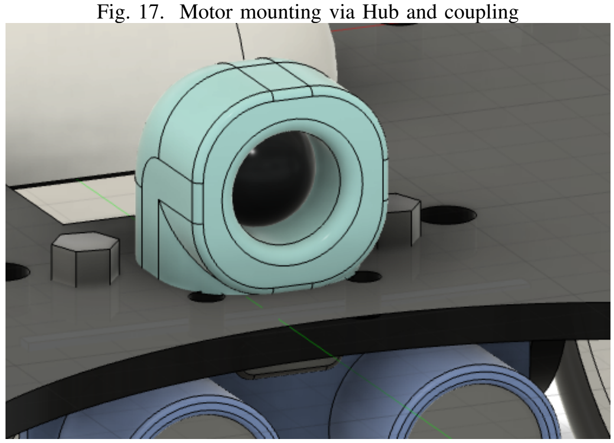

Fig. 18. 3D printed design for IR mount with heat shrink screws

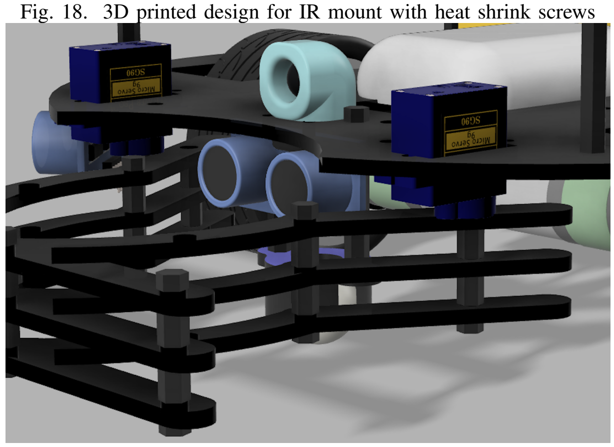

Fig. 19. Servo mounting and press fit design for gripper attachments

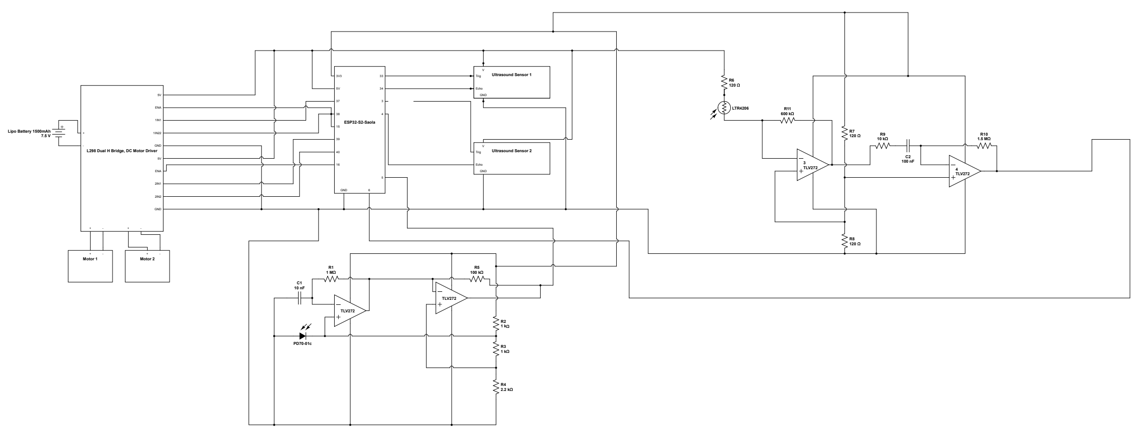

Fig. 20. Entire Circuit

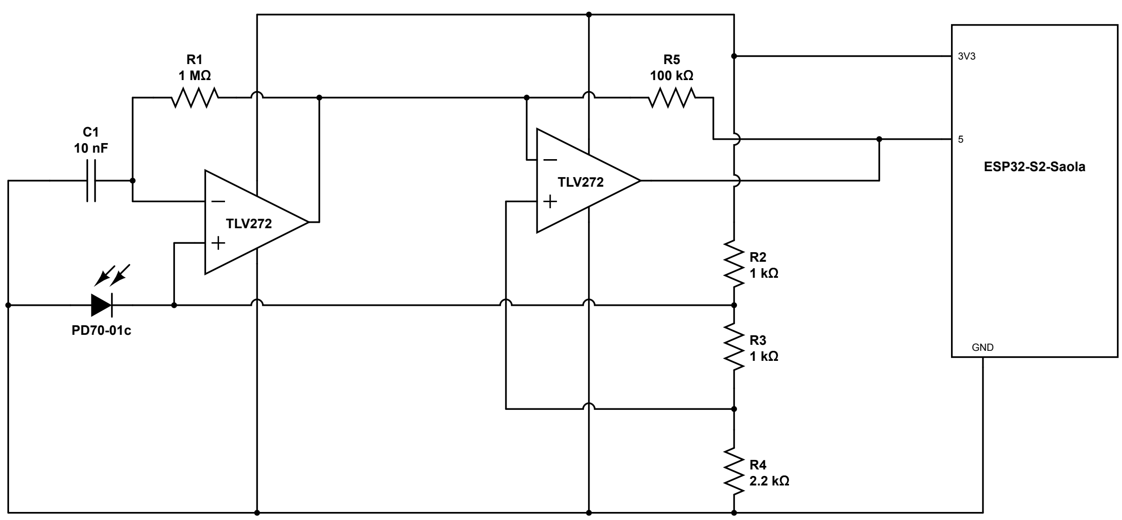

Fig. 21. Vive Circuit

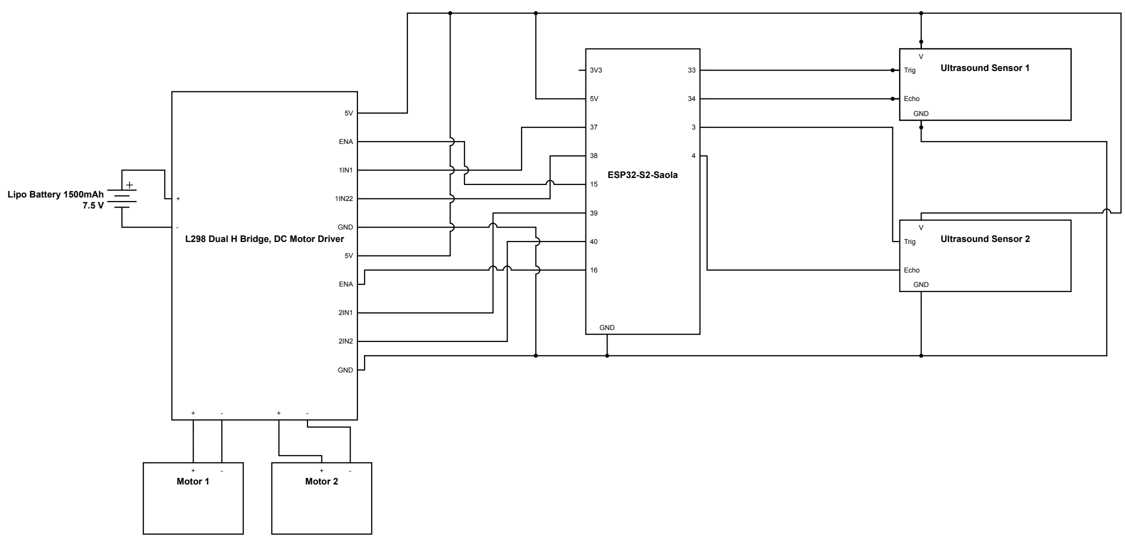

Fig. 22. Wallfollowing Circuit

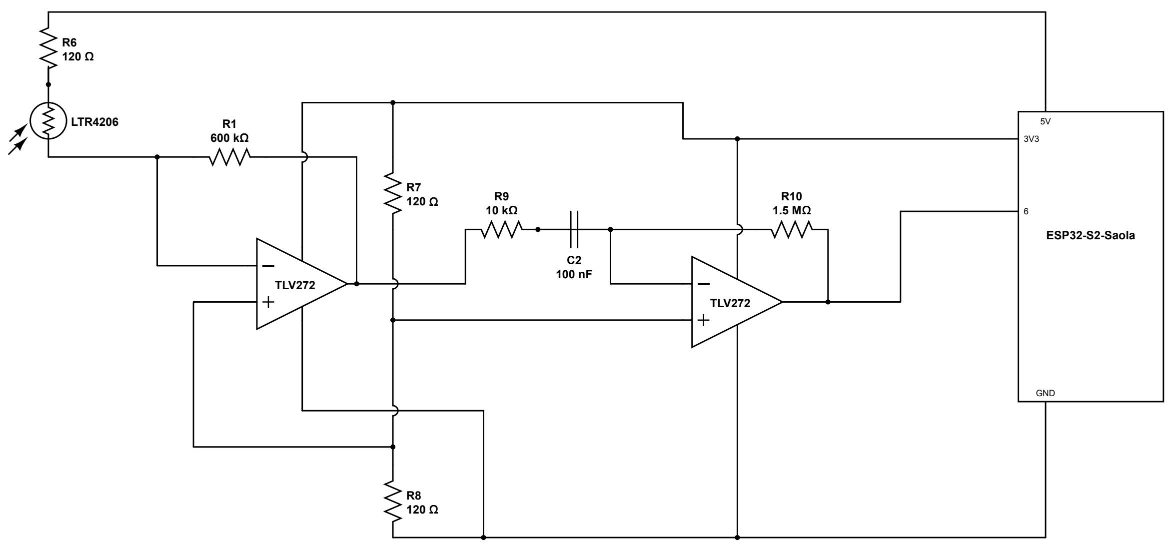

Fig. 23. Beacon Circuit

Videos:

- Wall Following
- Push Police Car
- Beacon Tracking