WRO 2024 - FUTURE ENGINEERS

Senku Team- Documentary

Ammar Daher (coach)
Hanna Hanna
Ritta Taleb
George Ateia

Contents

1 Mobility Management **5**
1.1 VEHICLE MAIN BODY: 5
1.1.1 3D Printing parts: 6
1.2 MOTION MECHANISM 8
1.2.1 Ackermann Steering Mechanism 8
1.2.2 Differential Gear 9
1.2.3 MOTOR DRIVER 11
1.2.4 Wheels 12
1.2.5 Lego Axles: 13
1.3 Engineering Principles 13

2 Power and Sense Management: **14**
2.1 Electrical Parts 14
2.2 Overall Circuit Current Usage 32
2.3 Circuit Grounding 33
2.4 Power Supply 33
2.5 Overall sheme 34
2.6 WIRING DIAGRAM 35
2.7 Bill of Materials 36
2.7.1 Assembly List 36
2.7.2 Shopping List 37
2.7.3 Notes 37

3 Obstacle Management **38**
3.1 OPEN CHALLENGE ALGORITHM 38
3.1.1 Moving at safe zone: 38
3.1.2 Moving into the Danger Zone: 39
3.1.3 Detecting turns and direction: 39
3.2 COMPUTER VISION: 41
3.3 IMAGE PROCESSING: 42
3.3.1 Setting the Camera Properties: 42
3.3.2 Pre Processing: 42
3.3.3 Masking: 43
3.3.4 Contouring: 44
3.4 Avoid Pillars Algorithm: 45
3.5 Perform of parallel parking: 46
3.6 Problems We Encountered: 47
3.6.1 Wave Interference: 47
3.6.2 Small Turn Angle and Sliding: 47
3.6.3 Flickering: 47

4 Suggestions and future modifications **48**
4.1 Open Challenge: 48
4.2 Obstacle Challenge: 49
4.3 Differential: 49
4.4 Micro-controller Upgrade: 49
4.5 Sensor Upgrades 50
4.5.1 Distance Measurement: 50
4.5.2 MPU Upgrade: 50
4.6 Power Management System (PMS) 50
4.7 Conclusion 50

5 Vehicle Photos **51**

6 Team Photo **53**

List of Figures

1 Ultrasonic Sensor Holder Base 6
2 Camera Holder Components 6
3 JSumo Tan Technic Gear 20 Tooth Double Bevel 7
4 Hexagonal Rod 7
5 Battery Holder 7
6 Ackermann Steering 9
7 Lego Differential Gear 10
8 Calculation of (N_{wheel}) with Gear Ratio and ($N_{\text{motorNoLoad}}$) 11
9 Basic-open loop control of motor speed 11
10 Wheel 12
11 Lego Axle 13
12 Arduino mega 2560 15
13 Arduino mega sensor shield v2 manual 16
14 Raspberry Pi 4 Model B 17
15 raspberry Camera Module 3 75 degrees 19
16 Cooling fan 21
17 BTS7960 22
18 XL4005 DC-DC Step-Down 23
19 MG996R Servo 25
20 Li-Poly RC battery 25
21 500rpm DC motor 26
22 JSumo 1000 RPM motor 28
23 JSumo Motor holder 28
24 MPU 6050 29
25 Ultrasonic sensors 31
26 Vehicle electronic - electrical scheme 34
27 Vehicle electronics: Wiring Diagram 35
28 zones Division 38

29 Open challenge flowchart . . . 40
30 RGB to HSV . . . 42
31 Red 1 Mask Analysis . . . 43
32 Red 2 Mask Analysis . . . 43
33 Green Mask Analysis . . . 44
34 Contouring Process . . . 44
35 contouring Analysis . . . 45
36 Vehicle corner parking . . . 45
37 Obstacle challenge flowchart (for high quality image, click here). . . . 46
38 Angle Calculation . . . 48
39 Robot views 1 . . . 51
40 Robot views 2 . . . 52
41 Team Photo . . . 53

List of Tables

1 Specifications of 65mm Rubber Wheels . . . 12
2 Input Current Requirements for Senku's Self-Driving Car Components . . . 32
3 Assembly List . . . 36
4 Shopping List . . . 37

Introduction

The development and design process of an autonomous vehicle created for participation inWRO 2024 Future Engineers Category . The car represents a fusion of theoretical, mathematical, and mechanical engineering principles combined with complex programming tasks and computer vision techniques. The project's main scope is to design an autonomously controlled vehicle that is able to navigate its environment while realizing the overall theme of Earth Allies . In this report, we discuss the details of the project and the theoretical aspects that form the basis for it. We describe the mathematical principles on which the realization of both control laws is based, facilitating precise control and navigation of the vehicle. We also get into mechanical engineering, focusing on the physical parts and systems that enable the functions.Our GitHub repository. Furthermore, there are a lot of programming tasks and computer vision techniques used in this project. We will be checking out the algorithms and software implementations used to process visual information and take decisions in real time. We now bring these visions into the vehicle's operation, with features like computer vision, which enables the vehicle to perceive and interpret its environment, hence navigating safely and efficiently. We combine these diverse disciplines in our quest to build an autonomous vehicle that showcases engineering prowess. we aim to contribute to the development of autonomous vehicles and their vast potential to change mobility and connectivity worldwide.

1. Mobility Management

1.1 VEHICLE MAIN BODY:

The vehicle main body It was divided into two parts. 3D Printing Section This section consists of 4 parts: the upper structure, two parts of which form a camera holder, and the last part is a battery holder. The second part of the structure is an aluminum base due to its lightness and durability. The main Microcontroller is Atmega mounted on Arduino Mega 2560 board. To ensure the accurate and efficient control and direction of the vehicle's movement, we employed crucial components:

- Servo Motor Model
- DC JSumo Motor 1000 RPM
- BTS Motor Driver

The servo motor model was selected for controlling the vehicle's steering, while the BTS7960 motor driver is utilized to control the movement of the DC motor in order to provide the necessary power to drive the vehicle's motion.

1.1.1 3D Printing parts:

we utilized SOLIDWORKS to design the components, followed by 3D printing these designs using PLA material. This process enabled us to create precise and customized parts for our robot, ensuring optimal performance and adherence to our design specifications.

Ultrasonic Sensor Holder Base The ultrasonic sensor holder base is used to mount the ultrasonic sensor, providing a stable platform for accurate distance measurements.

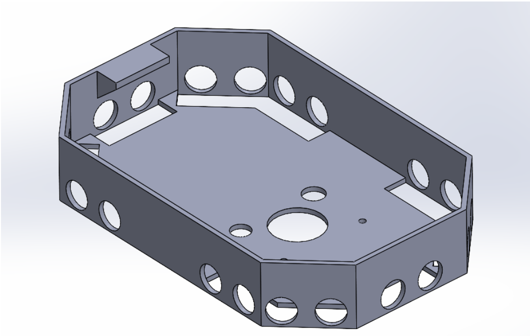
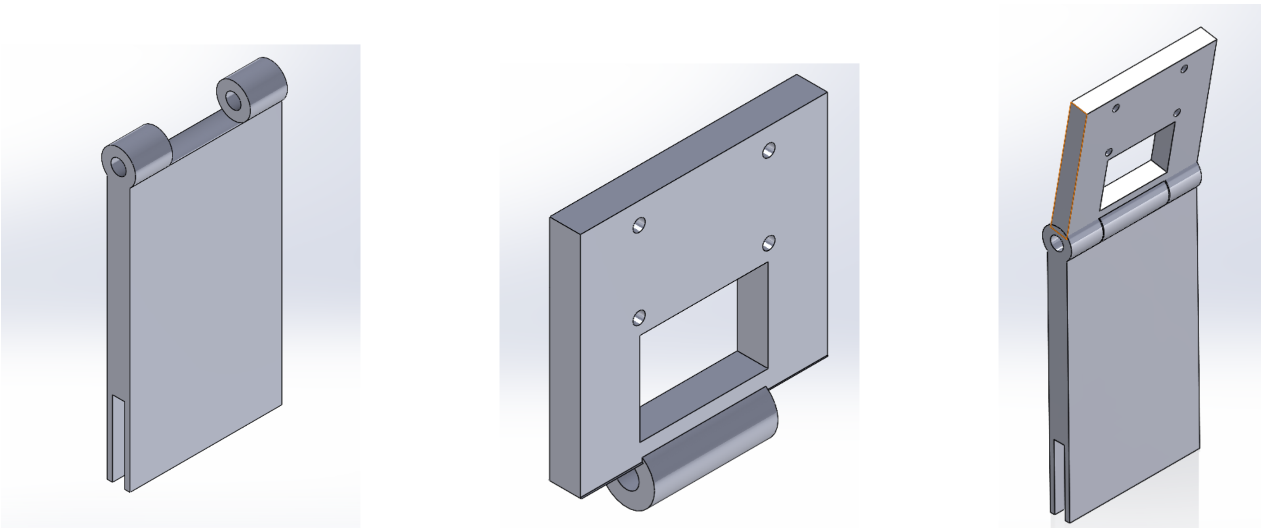


Figure 1: Ultrasonic Sensor Holder Base

Camera Holder The camera holder is designed to securely hold the camera in place, ensuring stability and precision in positioning. It includes both parts of the holder and the complete assembly.



Camera Holder Part 1

Camera Holder Part 2

Camera Holder Assembly

Figure 2: Camera Holder Components

agraphJSumo Tan Technic Gear 20 Tooth Double Bevel This gear is a crucial component in the assembly, providing the necessary transmission of motion through its double bevel design.

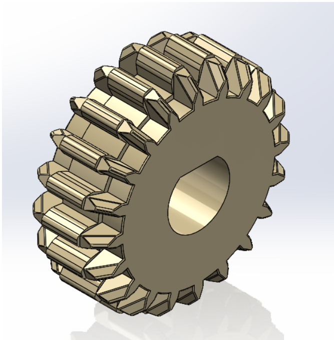


Figure 3: JSumo Tan Technic Gear 20 Tooth Double Bevel

Hexagonal Rod The hexagonal rod is a key component, used for providing structural support in the assembly. It features a hollow center which allows for integration with other parts.

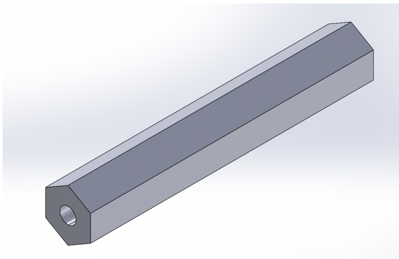


Figure 4: Hexagonal Rod

Battery Holder The battery holder is designed to securely hold the battery in place, providing necessary power to the assembly. It is robustly built to ensure stability and easy replacement of batteries when needed.

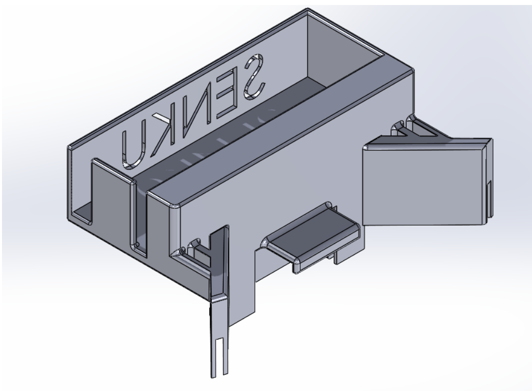


Figure 5: Battery Holder

1.2 MOTION MECHANISM

We used Ackermann steering and to achieve that a servo motor was installed in way that can move the steering mechanism freely, while the DC motor (1000 rpm) serves as the driving motor. The kinetic energy generated by the DC motor is transferred to the back wheels through a differential gearbox. The wheels receive this energy via a drive shafts of LEGO components.

1.2.1 Ackermann Steering Mechanism

Our design incorporates an Ackermann steering mechanism, which is crucial for ensuring efficient and responsive steering performance. The Ackermann steering principle helps to reduce tire wear and improve the vehicle's handling, particularly during turns. Here are the key features and benefits of the Ackermann steering system:

Enhanced Maneuverability The Ackermann steering geometry is designed to minimize tire slippage during turns. This is achieved by arranging the steering linkages so that the inner wheel turns at a sharper angle than the outer wheel, allowing each wheel to follow its own natural arc around the corner. This results in more precise steering and improved vehicle control.

Simplified Design The Ackermann steering mechanism simplifies the overall steering system design. By eliminating the need for complex gear arrangements, it reduces the weight and potential points of failure within the steering system. This simplicity also translates into easier maintenance and lower manufacturing costs.

Working Mechanism of Ackermann Steering The functionality of the Ackermann steering system can be summarized as follows:

1. **Steering Input Translation:** When Servo motor is turned, the input is transmitted through the steering linkage to both front wheels. Due to the specific angles and lengths of the steering arms, the inner wheel turns more sharply than the outer wheel.
2. **Cornering:** During cornering, the Ackermann geometry ensures that the wheels follow their respective paths, minimizing slippage and improving the vehicle's grip on the play field.

Key Parameters of Ackermann Steering

Steering Ratio The steering ratio in the Ackermann steering system describes the relationship between the rotation of Servo angle and the resulting rotation of the wheels. It can be expressed mathematically as:

$$\text{Steering Ratio} = \frac{\theta_{\text{Servo angle}}}{\theta_{\text{wheels}}}$$

A higher steering ratio means that a smaller rotation of Servo angle results in a larger rotation of the wheels, providing greater turning capability. Conversely, a lower steering ratio requires more rotation of Servo to achieve the same amount of wheel rotation, resulting in a less sensitive steering response.

Servo Motor Torque Steering torque (T_{servo}) is the force applied to the servo as a result of the torque output from the steering system. It is calculated by multiplying the applied force by the effective lever arm length (L_{eff}):

$$T_{\text{servo}} = F_{\text{steering}} \times L_{\text{eff}}$$

Higher sevo motor torque indicates that more force is needed to turn the wheels, providing greater resistance to steering and a more stable driving experience.

Steering Wheel Angle The steering wheel angle refers to the angular displacement of the steering wheel from its neutral position. This angle determines the orientation of the vehicle's steering mechanism and, consequently, its direction of movement:

$$\theta_{\text{servo}} = \theta_{\text{Sevo angle}} \times \text{Steering Ratio}$$

Sevo motor angle directly influences the turning radius of the vehicle, affecting its maneuverability and handling characteristics.[14].

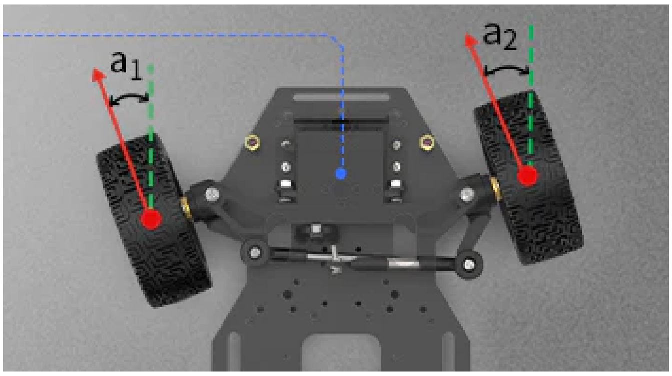


Figure 6: Ackermann Steering

1.2.2 Differential Gear

We used Lego differential gear, The Lego differential gear is a vital component in our robot that ensures the rear wheels can spin at different rates while maintaining balanced power distribution. This mechanism is particularly crucial during turns or when traction varies between the two wheels, allowing for smooth and controlled movement. Here's an overview of the differential gear system and its components:

Components

- **1. Ring Gear:** This large gear, attached to the axle shaft, receives power from the engine or driveshaft.
- **2. Pinion Gear:** A smaller gear that connects to the drive shaft and meshes with the ring gear.
- **3. Side Gears:** Two gears that link the differential to the axle shafts.
- **4. Spider Gears:** Positioned between the side gears, these gears permit the wheels to rotate at varying speeds.

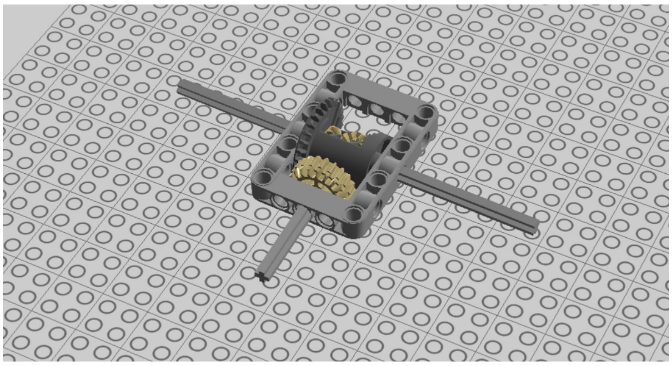


Figure 7: Lego Differential Gear

Functionality When the vehicle travels in a straight line, the differential distributes torque equally to both wheels. However, during a turn, the inside wheel (closer to the center of the turn) needs to rotate slower than the outside wheel to prevent slippage. The differential achieves this by allowing the side gears to rotate at different speeds while still distributing power effectively.

Calculating Gear Ratio Before designing a differential gear, determining the gear ratio is crucial. This involves understanding the motor specifications and the differential's mechanical setup. The gear ratio can be calculated by counting the teeth on the ring gear and the side gears:

- Let R be the number of teeth on the ring gear.
- Let $S1$ and $S2$ be the number of teeth on each side gear.

The gear ratio (GR) is given by the formula:

$$GR = \frac{R}{S1} + \frac{R}{S2}$$

This ratio indicates the rotational speed and torque distribution between the two side gears of the differential. Once calculated, this ratio helps relate the motor speed to the wheel speed. The wheel speed (N_{wheel}) can be derived from the motor's no-load speed ($N_{\text{Motor, no-load}}$) using the formula:

$$N_{\text{wheel}} = \frac{N_{\text{Motor, no-load}}}{GR}$$

The wheel speed reflects the rotational speed of the differential's side gears, directly influencing the vehicle's speed. While theoretical calculations provide a baseline, real-world values often require calibration due to various unaccounted constraints.[15].

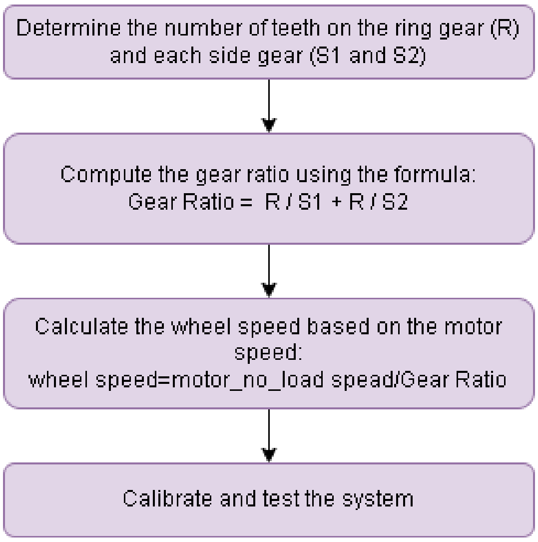


Figure 8: Calculation of (N_{wheel}) with Gear Ratio and ($N_{\text{motorNoLoad}}$)

1.2.3 MOTOR DRIVER

A motor driver is an indispensable device in controlling motors because it allows the safe handling and delivery of required power for driving the motor; at the same time, it includes protection features to guard against damage under abnormal operating conditions. In addition, the motor driver provides accurate control, speed, direction, and acceleration. A driver is a must for those applications that require precise positioning or motion control, as in our case. It has bidirectional control, works with various motor types, and optimizes power usage for more efficiency.

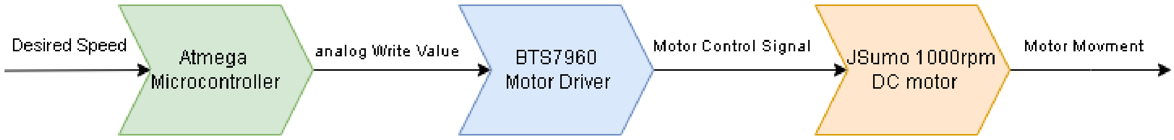


Figure 9: Basic-open loop control of motor speed

1.2.4 Wheels

Wheels were used in the robot's design, their advantage comes from their efficiency and light weight. These wheels provide good friction, which enhances stability and maneuverability on various surfaces. [26]

Figure 10: Wheel

Specification	**Details**
Diameter	65mm
Width	25-30mm
Material	Rubber tire with plastic or metal hub
Axle Hole	Compatible with standard hex couplings (often 4mm or 6mm)
Coupling Type	Lengthen copper coupling with hex connection
Weight	Usually light, suitable for small to medium-sized RC vehicles
Use	Suitable for DIY projects, Arduino smart car chassis, and other robotics applications

Table 1: Specifications of 65mm Rubber Wheels

1.2.5 Lego Axles:

Using the LEGO axles (4.8mm Diameter) on both the differential gears mechanism and the wheels connecting leads to a significant improvement in the robot's efficiency. The axles deliver accurate rotary motion while cutting down friction and energy losses. It is made of PA (Polyamide), which is a type of material that can withhold high loads and impacts. This means it is suitable for elements that have to be tough and can interact with each other, thus enabling a smooth power transmission of prolonged operation.

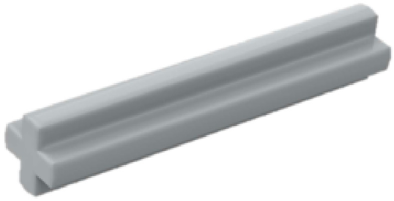


Figure 11: Lego Axle

1.3 Engineering Principles

The selection of materials and components for our vehicle was guided by several key engineering principles:

- **Torque and Speed Calculations:** The motors were chosen based on the required torque to overcome friction and inertia, ensuring sufficient speed for the competition.
- **Material Selection:** Aluminum was selected for its strength-to-weight ratio, providing a lightweight yet durable structure.
- **3D printing selection**: PLA is ease of Printing, Strength and aligns with sustainable engineering practices, making it an eco-friendly choice.

2. Power and Sense Management:

2.1 Electrical Parts

Arduino Mega 2560 : The Arduino Mega 2560 is a powerful and versatile open-source microcontroller board, making it a popular choice for complex and resource-intensive projects. It is based on the ATmega2560 microcontroller, which provides ample processing power and memory to handle a wide range of applications. Below are the essential features and specifications of the Arduino Mega 2560:

1. **Microcontroller:**

- **ATmega2560:** The heart of the Arduino Mega 2560 is the ATmega2560 microcontroller. It includes:
  - **256 KB of Flash Memory:** Used for storing the program code.
  - **8 KB of SRAM:** For data storage during program execution.
  - **4 KB of EEPROM:** For storing long-term data that remains even when the board is powered off.

2. **Digital I/O Pins:**

- The board features **54 digital input/output pins**, which provide a vast array of connectivity options.
- Out of these, **15 pins can be used as PWM (Pulse Width Modulation) outputs**, useful for tasks like dimming LEDs or controlling motors.

3. **Analog Inputs:**

- There are **16 analog input pins** on the Mega 2560.
- These pins have a **10-bit resolution**, meaning they can distinguish between 1024 different voltage levels, ideal for reading sensors and other analog devices.

4. **Communication Interfaces:**

- The Mega 2560 supports multiple communication interfaces, making it highly versatile:
  - **UART (Universal Asynchronous Receiver/Transmitter):** For serial communication.
  - **SPI (Serial Peripheral Interface):** For high-speed data transfer between the microcontroller and peripherals.
  - **I2C (Inter-Integrated Circuit):** For communication with various sensors and modules.

5. **USB Interface:**

- The board includes a **USB interface**, which facilitates easy connection to a computer for programming and serial communication.
- The USB interface can be configured to appear as a virtual serial port, enabling seamless integration with various software development environments.

6. **Operating Voltage:**

- The Arduino Mega 2560 operates at **5V**, ensuring compatibility with a wide range of sensors, modules, and other electronic components.

7. **Compatibility:**

- The board is fully compatible with the **Arduino software (IDE)** and programming language, making it straightforward for beginners and advanced users to start coding and prototyping.
- It also supports a vast ecosystem of **Arduino libraries**, which can significantly speed up development time by providing pre-written code for many common tasks.

8. **Shield Compatibility:**

- Designed to be compatible with all standard Arduino shields, the Mega 2560 can be expanded to include additional functionality.
- Shields can add capabilities like **Wi-Fi**, **Bluetooth**, **motor control**, and more, simply by stacking them on top of the board.

Given its extensive I/O capabilities and robust communication options, the Arduino Mega 2560 is ideally suited for a wide range of applications, including but not limited to: robotics, home automation, data acquisition, and complex prototyping. In summary, the Arduino Mega 2560 is a powerful microcontroller board that offers significant flexibility and performance for complex and demanding projects. Its rich set of features and extensive compatibility with shields and the Arduino software ecosystem make it an excellent choice for both hobbyists and professionals.[5]

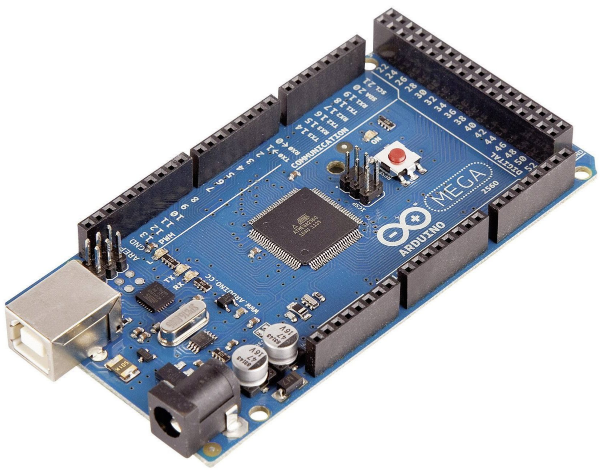


Figure 12: Arduino mega 2560

arduino sensor shield v2.0 The Sensor Shield v2.0 Mega is an expansion board designed for the Arduino Mega 2560. It provides a convenient way to connect various sensors, actuators, and modules to the Arduino Mega, simplifying the prototyping process. Key features of the Sensor Shield v2.0 Mega include:

- **Plug-and-Play:** Offers easy connection of sensors and modules without the need for soldering.
- **Multiple Interfaces:** Supports I2C, UART, and analog interfaces, making it versatile for different types of components.
- **Compatibility:** Fully compatible with Arduino Mega 2560 and other compatible boards.
- **Pin Headers:** Provides standard pin headers for easy and secure connections.

The benefits of using the Sensor Shield v2.0 Mega include:

- **Ease of Use:** Simplifies the process of connecting sensors and modules, reducing setup time.
- **Versatility:** Supports a wide range of interfaces, allowing for diverse project applications.
- **Compatibility:** Works seamlessly with Arduino Mega 2560, ensuring smooth integration with existing projects.
- **Convenience:** Reduces the need for soldering and manual wiring, making prototyping faster and more reliable.

For more details, you can visit the product page here[25].

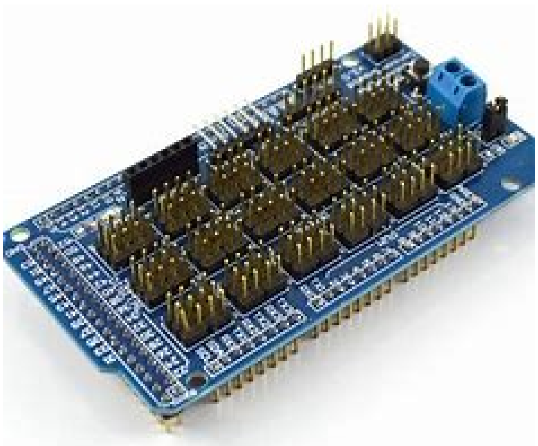


Figure 13: Arduino mega sensor shield v2 manual

Raspberry Pi 4 Model B: The Raspberry Pi 4 Model B features a high-performance 64-bit quad-core processor, dual-display support at resolutions up to 4K via a pair of micro HDMI ports, hardware video decode at up to 4Kp60, up to 8GB of RAM, dual-band 2.4/5.0 GHz wireless LAN, Bluetooth 5.0, Gigabit Ethernet, USB 3.0, and PoE capability (via a separate PoE HAT add-on). For the end user, Raspberry Pi 4 Model B provides desktop performance comparable to entry-level x86 PC systems.

This product retains backwards compatibility with the prior-generation Raspberry Pi 3 Model B+ and has similar power consumption, while offering substantial increases in processor speed, multimedia performance, memory, and connectivity.

The dual-band wireless LAN and Bluetooth have modular compliance certification, allowing the board to be designed into end products with significantly reduced compliance testing, improving both cost and time to market.

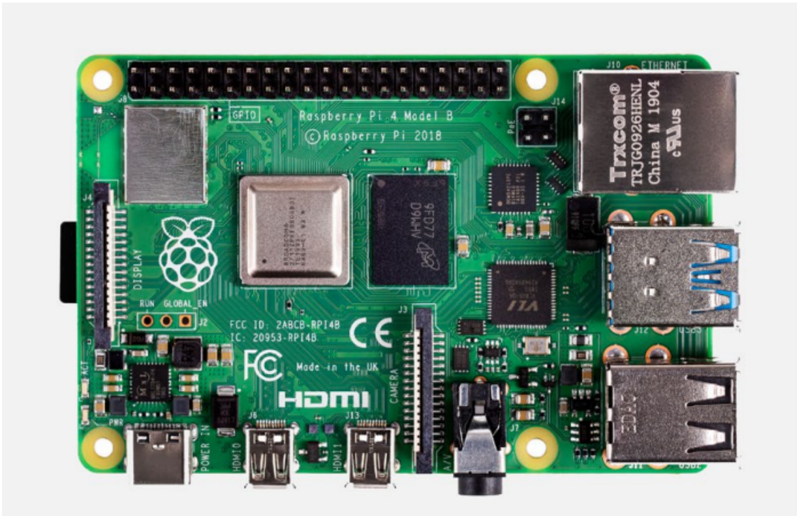


Figure 14: Raspberry Pi 4 Model B

Specification:

- **Processor:** Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
- **Memory:** 1GB, 2GB, 4GB or 8GB LPDDR4 (depending on model) with on-die ECC
- **Connectivity:**
  - 2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN, Bluetooth 5.0, BLE
  - Gigabit Ethernet
  - 2 × USB 3.0 ports
  - 2 × USB 2.0 ports
- **GPIO:** Standard 40-pin GPIO header (fully backwards-compatible with previous boards)
- **Video & sound:**
  - 2 × micro HDMI ports (up to 4Kp60 supported)

- 2-lane MIPI DSI display port
- 2-lane MIPI CSI camera port
- 4-pole stereo audio and composite video port

- **Multimedia:**
  - H.265 (4Kp60 decode)
  - H.264 (1080p60 decode, 1080p30 encode)
  - OpenGL ES, 3.0 graphics
- **SD card support:** Micro SD card slot for loading operating system and data storage
- **Input power:**
  - 5V DC via USB-C connector (minimum 3A)
  - 5V DC via GPIO header (minimum 3A)
  - Power over Ethernet (PoE)–enabled (requires separate PoE HAT)
- **Environment:** Operating temperature 0–50ºC
- **Production lifetime:** Raspberry Pi 4 Model B will remain in production until at least January 2034.
- **Compliance:** For a full list of local and regional product approvals, please visit here.

Warnings

- This product should only be connected to an external power supply rated at 5V/3A DC or 5.1V/3A DC minimum. Any external power supply used with Raspberry Pi 4 Model B shall comply with relevant regulations and standards applicable in the country of intended use.
- This product should be operated in a well-ventilated environment and, if used inside a case, the case should not be covered.
- This product should be placed on a stable, flat, non-conductive surface in use and should not be contacted by conductive items.
- The connection of incompatible devices to the GPIO connection may affect compliance and result in damage to the unit and invalidate the warranty.
- All peripherals used with this product should comply with relevant standards for the country of use and be marked accordingly to ensure that safety and performance requirements are met. These articles include but are not limited to keyboards, monitors, and mice when used in conjunction with Raspberry Pi.
- Where peripherals are connected that do not include the cable or connector, the cable or connector must offer adequate insulation and operation in order that the relevant performance and safety requirements are met.

Safety Instructions To avoid malfunction or damage to this product, please observe the following:

- Do not expose to water, moisture, or place on a conductive surface whilst in operation.
- Do not expose to heat from any source; Raspberry Pi 4 Model B is designed for reliable operation at normal ambient room temperatures.
- Take care whilst handling to avoid mechanical or electrical damage to the printed circuit board and connectors.
- Avoid handling the printed circuit board whilst it is powered and only handle by the edges to minimize the risk of electrostatic discharge damage.[18]

raspberry Camera Module 3 :

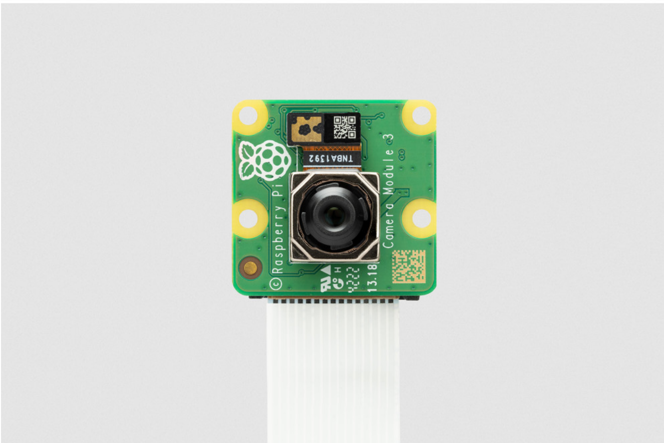


Figure 15: raspberry Camera Module 3 75 degrees

Specification:

- **Still resolution:** 11.9 Megapixels
- **Sensor:** Sony IMX708
- **Sensor resolution:** 4608 x 2592 pixels
- **Sensor image area:** 6.45 x 3.63 mm (7.4 mm diagonal)
- **Pixel size:** 1.4 μm x 1.4 μm
- **Optical size:** 1/2.43"

- **Focus:** Motorized
- **Depth of field:** Approx 5 cm to ∞
- **Focal length:** 2.75 mm
- **Horizontal Field of View (FoV):** 102 degrees
- **Vertical Field of View (FoV):** 67 degrees
- **Focal ratio (F-Stop):** F2.2
- **Maximum exposure times:** 112 seconds
- **Size:** Around 25 x 24 x 12.4 mm
- **Weight:** 4g
- **Video modes:**
  - 2304 x 1296p56
  - 2304 x 1296p30 HDR
  - 1536 x 864p120
- **NoIR version available:** No

The Camera Module 3 (75-degree FoV) is a great fit for obstacle detection as well because;

1. **Flexibility**: The Camera Module 3 could be fine-tuned for image size, ISO (light sensitivity), frame rate, and shutter speed. This flexibility guarantees high-quality pictures under various lighting conditions, something very important for reliable outcomes in computer vision and sensor fusion jobs.
2. **Perfect Field of View**: The 75-degree field provides a natural view that distorts very little. It covers so much ground it's perfect to get the whole track and features on the trail. The midrange (35 mm) camera strikes the difference between wide and tight views with a view you should be able to use on both close-up subjects or from farther away.
3. **In-lens correction**: In-camera will not compensate as much fisheye effect, which leads to a more clear and skewed picture. For accurate image processing and obstacle detection that is necessary to lead the autonomous navigation of the robot.
4. **Versatile**: The Camera Module 3 is built to meet the specific requirements of a variety of projects, making it suitable for use in domestic monitoring applications as well as business and robotic solutions. This makes it a perfect choice for the wide range of difficult tasks from following signaling lights through properly recognizing and responding towards obstacles in WRO competition.
5. **Seamless integration**: CameraModule3 can interface well with the other sensors and components on-board the robotic vehicle due to high coverage, and good quality imaging capabilities. This integration helps in making the robot fully automatic which improves the performance and reliability of the robot during the competition.

In conclusion, the Camera Module 3 with a field of view (FoV) set to 75 degrees is a well-rounded and robust option for the WRO Challenge in year-round settings offering the functionality required as much flexibility needed to succeed at this competition.[20]

Cooling fan: It is used to reduce the heat generated by Raspberry Pi microprocessor, thus reducing overheating and increasing the performance of the CPU.

Figure 16: Cooling fan

BTS7960 Motor Driver : A standard module of the dual H-bridge motor driver, BTS7960, allows one to control the speed and direction of DC motors. It is often applied in robotics, automation, and other projects where precision matters in handling motors. The Main Features and Specifications of the BTS7960 Motor Driver:

- **H-Bridge Configuration:** An H-Bridge configuration is used in the BTS7960 Motor Driver. It is designed with four power MOSFETs (Metal-Oxide-Semiconductor Field-Effect Transistors) placed in pairs. This enables bidirectional control of the motor—it can drive the motor either forward or reverse.
- **Motor Compatibility:** The motor driver module is dedicatedly used for DC motors and can support a broad range of motor voltages, typically between 5 V and 27 V. The maximum continuous current per channel typically remains around 43 A, making it very suitable for driving high-power motors.
- **Current Sensing:** The BTS7960 Motor Driver incorporates built-in current sensing. It comprises two current-sense pins such that the user can monitor the current flowing through each motor channel. It can be used in various applications, such as motor current protection or feedback control. item **PWM Control:** The driver supports motor control with PWM and varying speed. The speed of a motor can be very precisely adjusted by altering the duty cycle of the PWM signal. The module can receive a PWM input signal from the microcontroller or any other source.
- **Overcurrent and Overtemperature Protection:** The BTS7960 Motor Driver has built-in protection mechanisms for both the driver and the motor against overcurrent and overtemperature. These circuits help prevent damage to the motor driver module and the connected motor.
- **Input and Output Connections:** The motor driver module typically has separate input pins for controlling the direction and speed of a motor. These control pins can be connected to the digital output pins

of a microcontroller or other control devices. The motor connections are made to the driver's output terminals, allowing you to connect your DC motor.

- **Heat Dissipation:**Since the BTS7960 motor driver can handle high current, it may dissipate some heat during operation. This module often features an integrated heat sink or a mounting hole for fitting an external heat sink to dissipate the heat effectively.
- **Compatibility:**The BTS7960 Motor Driver can interface with most microcontrollers, such as Arduino, Raspberry Pi, or any other microcontroller capable of producing the necessary control signals for the driver module.[6]

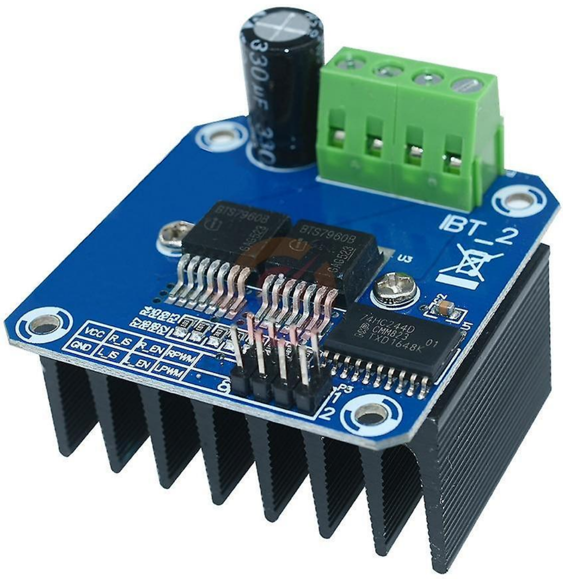


Figure 17: BTS7960

XL4005 DC-DC Step-Down: The XL4005 DC-DC Step-Down Power Module is a versatile and efficient non-isolated step-down converter, commonly used in various electronic projects and applications. This module is capable of converting a higher input voltage (ranging from 5V to 32V) to a lower, stable output voltage, which can be adjusted as per the requirements of the specific application. The output current capacity of the XL4005 module is up to 5A, making it suitable for powering a variety of devices and circuits. The key features and specifications of the XL4005 module include:

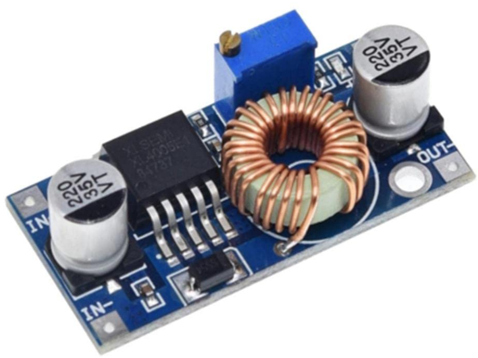


Figure 18: XL4005 DC-DC Step-Down

- **Input Voltage Range**: The module can accept input voltages from 5V to 32V, providing flexibility for use with different power sources.
- **Output Voltage Range**: The output voltage is adjustable from 0.8V to 30V, allowing for precise voltage regulation tailored to specific needs.
- **Output Current**: Capable of delivering up to 5A of current, the XL4005 module can power high-current devices efficiently.
- **Efficiency**: The conversion efficiency can reach up to 95%, minimizing power loss and heat generation during operation.
- **Thermal Performance**: The module includes thermal shutdown protection and a heat sink to maintain optimal performance and prevent overheating.
- **Short Circuit Protection**: Built-in protection against short circuits ensures the safety and reliability of the module and the devices it powers.

The XL4005 DC-DC Step-Down Power Module is a reliable and efficient solution for converting and regulating voltages in various electronic projects, ensuring a stable and adjustable power supply for diverse applications.[10]

The TowerPro MG996R servo : The TowerPro MG996R servo is a robust and versatile actuator used in various robotic and automation projects. This servo is well-regarded for its high torque and metal gear construction, making it suitable for demanding applications where durability and performance are critical. Below is an overview of its key features and specifications:

- **High Torque:** The MG996R servo offers a stall torque of 12.0 kg.cm at 6V, making it powerful enough for applications requiring significant force, such as steering mechanisms in remote-controlled vehicles or robotic arms.
- **Metal Gears:** The internal gears are made of metal, which enhances durability and reduces the likelihood of wear and tear compared to plastic gears. This makes the MG996R ideal for high-stress environments.
- **Wide Operating Voltage Range:** The servo operates within a voltage range of 4.8V to 7.2V, providing flexibility in various power supply configurations. This range allows the servo to be compatible with standard battery packs and DC power supplies used in robotics.
- **Speed and Precision:** At 6V, the MG996R can rotate 60 degrees in 0.2 seconds, offering a balance between speed and control. This precision is crucial for applications requiring accurate positioning.
- **Control System:** The servo utilizes a standard pulse width modulation (PWM) control, which is widely supported by microcontrollers such as Arduino and Raspberry Pi. This makes integration into projects straightforward and user-friendly.

Specifications

- **Operating Voltage:** 4.8V to 7.2V
- **Stall Torque:** 12.0 kg.cm (at 6V)
- **Operating Speed:** 0.2 sec/60 degrees (at 6V)
- **Dimensions:** 40.7 x 19.7 x 42.9 mm
- **Weight:** 55g
- **Connector Type:** JR (universal)

The TowerPro MG996R servo is a reliable and powerful choice for any project requiring high torque and durability. Its metal gears, wide voltage range, and ease of control make it a favorite among hobbyists and professionals alike. Whether for robotics, automation, or RC applications, the MG996R stands out as a robust and versatile actuator.[11]

Figure 19: MG996R Servo

Li-Poly RC batter The 11.1V – 2700mAh 35C Li-Poly RC battery is an advanced lithium polymer (Li-Poly) battery designed for high-performance applications such as remote control (RC) vehicles. This battery is known for its high discharge rate, lightweight, and compact size, making it ideal for applications where power and efficiency are critical.

Figure 20: Li-Poly RC battery

Specifications:

- **Voltage:** 11.1V
- **Capacity:** 2700mAh
- **Discharge Rate:** 35C
- **Dimensions:** 135 x 42 x 23 mm
- **Weight:** 190g

- **Connector Type:** XT60
- **Balancing Connector:** JST-XH

The high discharge rate of 35C allows for rapid power delivery, ensuring that the RC vehicle can achieve peak performance. This makes it particularly suitable for racing and other high-intensity activities where quick acceleration and sustained power are required. The battery is equipped with a JST-XH balancing connector to ensure that all cells charge evenly, preventing overcharging and extending the battery's lifespan. Additionally, the high-quality construction of the battery reduces the risk of swelling, overheating, and other common issues associated with Li-Poly batteries. For optimal performance and longevity, it is recommended to charge the battery using a compatible Li-Poly charger and to follow the manufacturer's guidelines on charging times and cycles. It is also important to store the battery in a cool, dry place and to avoid exposing it to extreme temperatures. This 11.1V – 2700mAh 35C Li-Poly battery is ideal for use in various RC applications, including RC cars, RC boats, RC airplanes, and drones. By providing a reliable and powerful energy source, this battery enhances the overall performance and enjoyment of RC activities.[13]

500rpm DC motor : The 500 RPM DC motor is a versatile and efficient motor commonly used in various applications requiring moderate speed and torque. This motor is particularly well-suited for small robots, buggies, and automation projects.

Key specifications include:

Figure 21: 500rpm DC motor

- **Rated Voltage:** 12V DC
- **No Load Speed:** 500 RPM
- **Stall Torque:** 2.5 kg.cm
- **Rated Torque:** 1.2 kg.cm
- **No Load Current:** 0.2A

- **Stall Current:** 2.5A

Mechanical properties are as follows:

- **Motor Shaft Diameter:** 6 mm
- **Shaft Length:** 20 mm
- **Motor Body Length:** 50 mm
- **Motor Diameter:** 37 mm
- **Weight:** 150 grams

Electrical connections are as follows:

- **Power Wires:** Typically red (positive) and black (negative) wires for power connections.
- **Control:** Can be controlled using PWM signals from microcontrollers for precise speed regulation.

Performance graphs include:

- **Torque vs. Speed:** Displays the relationship between torque and speed, indicating how torque decreases as speed increases.
- **Current vs. Speed:** Shows the current draw at different speeds, highlighting the efficiency of the motor under various loads.

  This 500 RPM DC motor provides a reliable and efficient solution for projects requiring moderate speed and torque, making it an excellent choice for a wide range of applications from robotics to automation systems.

  JSumo 1000 RPM motor: This 1000 RPM motor from JSumo is a high-power version of another popular model, the Titan DC Gear Motor.

  Specifications:

  - **Voltage:** 12V
  - **RPM (no-load speed):** 1000
  - **Current:** 240mA (no-load), 5.9A (stall current)
  - **Torque:** 7.5Kg-cm (stall torque)
  - **Size:** Diameter: 37mm, Length: 75mm
  - **Shaft Size:** 6 mm with shaft length: 17 mm
  - **Gear Material:** Metal gear
  - **Mounting:** Fixed holes on front side, M3 screws

  The powerful motors are designed for heavy-duty applications or sumo robot models, providing durability and robust performance under load conditions. More details are available at JSumo.[21]

Figure 22: JSumo 1000 RPM motor

JSumo motor holder: The JSumo 1000 RPM DC motor holder is a vital component designed using SolidWorks and printed using PLA (Polylactic Acid) material for use in robotics and mechatronics projects. The holder is engineered to securely mount a DC motor, specifically one with a 1000 RPM rating, ensuring optimal performance and stability.

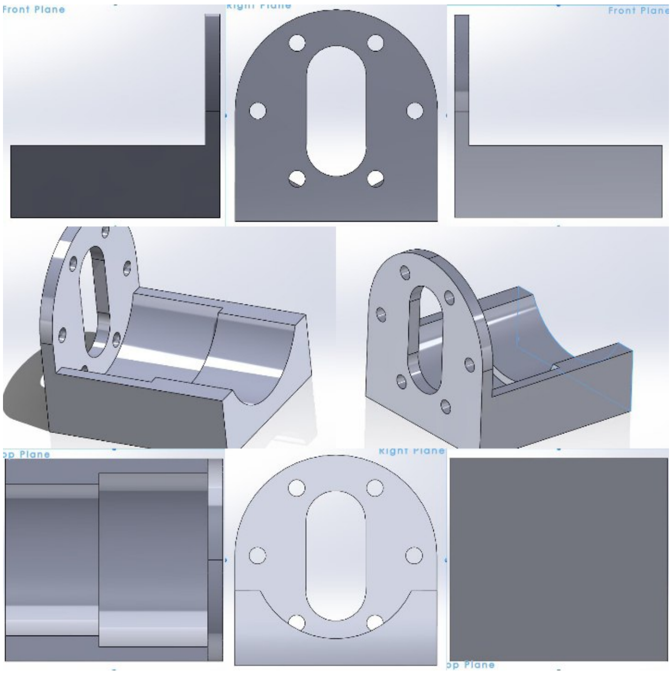


Figure 23: JSumo Motor holder

MPU-6050 : is a popular and widely used integrated circuit (IC) that combines a 3-axis gyroscope and a 3-axis accelerometer in a single package. It is commonly used in various applications such as motion sensing, orientation tracking, gesture recognition, and stabilization control systems.

The key features and specifications of the MPU-6050:

1. **3-Axis Gyroscope**: The MPU-6050 includes a 3-axis gyroscope, which measures angular velocity around three axes (X, Y, and Z). It provides accurate motion sensing capabilities and can be used to detect rotational movements.

2. **3-Axis Accelerometer**: The MPU-6050 also incorporates a 3-axis accelerometer, which measures linear acceleration along the three axes (X, Y, and Z). It enables the detection of changes in velocity and the determination of tilt or inclination.

3. **Digital Motion Processing**: The MPU-6050 features an onboard digital motion processor (DMP) that offloads motion processing tasks from the main microcontroller. The DMP performs sensor fusion, combining data from the gyroscope and accelerometer to provide accurate and reliable motion tracking information.

4. **I2C Interface**: The MPU-6050 communicates with the microcontroller or other devices using the I2C (Inter-Integrated Circuit) protocol. It has an I2C interface that allows for easy integration and control with various microcontrollers and development boards.

5. **Motion Detection Interrupts**: The MPU-6050 includes built-in motion detection capabilities and supports programmable interrupts. This allows the IC to generate interrupts to the microcontroller when predefined motion or orientation thresholds are met, enabling efficient and responsive operation.

6. **Low Power Consumption**: The MPU-6050 is designed to operate with low power consumption, making it suitable for battery-powered applications and devices with power constraints.

7. **Applications**: The MPU-6050 is widely used in applications such as robotics, drones, gaming controllers, virtual reality (VR) and augmented reality (AR) devices, inertial navigation systems, motion-controlled user interfaces, and many other projects where motion sensing and orientation tracking are required.[7]

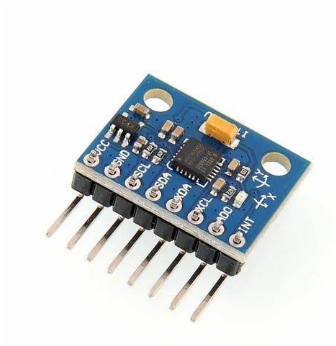


Figure 24: MPU 6050

HC-SR04 Ultrasonic Sensor The HC-SR04 is one of the popular ultrasonic distance sensor modules. The HC-SR04 is a sensor that uses ultrasonic waves to measure the distance by measuring how long it takes for a sound wave to reach and bounce from the target.

Tech Specs:

- **Working Voltage**: 5V DC
- **Current Output Level**: 15mA
- **Working Distance Range**: min. 4.23 inches (2 cm) - max. 177 inches (450 cm)
- **Resolution**: 0.12 inches (3 mm)
- **Characteristic Angle**: 15°
- **Communication Signal Frequency**: 40 kHz
- **Trigger Input Signal**: 1–10 μs High Pulse
- **Echo Output**: outputs a time-dependent state from the period how long it lasts at high state (even longer than maximal detection)

Applications:

- Home Automation and Process Control
- Factory Automation Systems
- Security Alarms
- Proximity Sensing
- Object Collision Alert
- User Interface System Integration

Tech Benefits:

- Wide Operation Voltage Range: The HC-SR04 works standardly on the fixed V supply, which helps it fit into various schematics using Arduino, Raspberry Pi, etc.
- **Interface**: It has an easy interface utilizing 4 pins Vcc, Gnd, Trigger, and Echo.
- **Distance**: Able to measure from 2 cm up to 450 cm
- **Low Power**: Operational current of 15 mA only, which suits battery-powered devices.
- **High Precision** : Stable distance measurement with a resolution of around 0.3 cm

The 10 sensors were mounted as follows:

- **L1 Sensor**: Left Front Closer to the frontend of the robot).
- **L2 Sensor**: Left Back Closer to the backend of the robot).
- **R1 Sensor**: Right Front Closer to the frontend of the robot).
- **R2 Sensor**: Right Back Closer to the backend of the robot).
- **F Sensor**: Frontal Sensor In the middle of the frontend of the robot).
- **B Sensor**: Rear Sensor In the middle of the backend of the robot.
- **BL Sensor**: Back Left Sensor - mounted on the edge of the vehicle and tilted by 45 degrees counterclockwise about the x-axis of the robot.
- **BR Sensor**: Back Right Sensor - mounted on the edge of the vehicle and tilted by 45 degrees clockwise about the x-axis of the robot.
- **FL Sensor**: Front Left Sensor - mounted on the edge of the vehicle and tilted by -45 degrees counterclockwise about the x-axis of the robot.
- **FR Sensor**: Front Right Sensor - mounted on the edge of the vehicle and tilted by -45 degrees clockwise about the x-axis of the robot.[8]

Figure 25: Ultrasonic sensors

2.2 Overall Circuit Current Usage

There are many reasons why it is important to have an accurate knowledge of the electrical component requirements which include:

- **Dual Mode (Operational and Sleep)**: Program the power supply system to provide different current requirements for robot electrical components in operational vs. sleep mode. Such that the power source selected is able to handle sufficient current at appropriate voltage levels to keep the system running stable with all performance.
- **Optimize for Battery Life**: If you kind of picture battery life as a pie, then understanding the power consumption of each slice is crucial to ensuring your app stays on with other apps at play. Proper predictive analysis such as the current draw estimates for different operational states will assist in formulating power-saving methods, selecting appropriate batteries, and hence this would increase the time of operation which is a good method to reduce constant replacement or recharging of batteries.
- **Heat Management**: Electrical components can start to overheat when being used at a higher current for more extended periods. Understanding the total load on a circuit helps to know how much heat is produced, so effective cooling measures can be undertaken before network under-performance or equipment breakdown occurs.
- **Safety**: Safe use or the most current draw that may occur is built into calculating tolerance levels in this section, including a brief bit about `power.py` and how it interfaces with related classes to handle current max values without firing bad things off electrically.
- **Cost Efficiency**: Knowledge is power, and knowing what each component needs in terms of electricity supply enables us to select cost-effective parts meeting the necessary specifications for performance and reliability.

Input current for components of the robot's circuit is shown in the table below:

Component	Input Current (mA)
Ultrasonic Sensor	15
MPU6050 Gyroscope Sensor	36
Raspberry Pi Model B4	3000
Arduino Mega	500
Servo Motor MG995R	500-2500 (depends on load)
Raspberry Pi Camera	250-300
BTS7960 Motor Driver	240-5900 (depends on load)

Table 2: Input Current Requirements for Senku's Self-Driving Car Components

Component Details:

- **Ultrasonic Sensor**: Typically requires between a few milliamperes (mA) to around 15 mA, depending on the specific model and operating conditions.

- **MPU6050 Gyroscope Sensor**: Requires approximately 36 mA.
- **Raspberry Pi Model B4**: Typically requires around 2.5 A, but it is recommended to use a power supply capable of providing 3 A for stable operation, considering connected peripherals and workload.
- **Arduino Mega**: Typically draws around 50 mA when idle, but current consumption can increase with additional peripherals or more complex programs.
- **Servo Motor MG995R**: Input current can vary between 500 mA and 2.5 A, depending on the load and specific model.
- **Raspberry Pi Camera**: Typically requires around 250-300 mA, depending on the model and settings.
- **JSumo DC motor 1000rpm**: No load current is 240mA, and stall curent is 5900A.

Design and development of a robust, high-performance power supply system optimized for size and cost require an understanding of these demands.

2.3 Circuit Grounding

Circuit grounding connects all components with a common ground, providing a reference point for all voltages in the circuit and dramatically simplifying the measuring of voltage at any point in the circuit. It ensures the signal moves accurately without distortion, prevents electrical shocks, and safeguards the equipment, as well as people, from hazards that can arise otherwise.

Grounding reduces electromagnetic interference (EMI) and radio frequency interference (RFI) by allowing a path for unwanted currents to dissipate harmlessly into the ground. This can greatly improve the performance and overall reliability of the circuit. Grounding stabilizes the voltage levels, preventing voltage fluctuation by ensuring a consistent electrical potential across all components. It may also help in reducing some noise and interference within the circuit. When several sections of the circuit share a common reference point, one can better filter noise and interference.

2.4 Power Supply

In the revised setup, we used an 11.1V – 2700mAh 35C Li-Poly RC battery as the power source, which directly provides 11.1V to both the BTS motor driver and the RioRand step-down DC-DC converter. The RioRand converter adjusts the voltage from 11.1V down to 5V to power the Arduino board. Additionally, we utilized the Arduino Mega Sensor Shield, which conveniently offers the GND and VCC pins for connecting ultrasonic sensors, a servo motor, and a gyroscope without the need for a bridge or extra pins. This configuration results in a visually pleasing circuit that provides easy access to all pins and simplifies the troubleshooting of faults and issues. The DC motor receives power directly from the battery through the BTS motor driver, maintaining the supply at 11.1 volts.

2.5 Overall sheme

Due to the large number of electronic components used in our circuit we used a special power supply distribution using basic electronic components. The following figure illustrates the electrical design of the circuit:

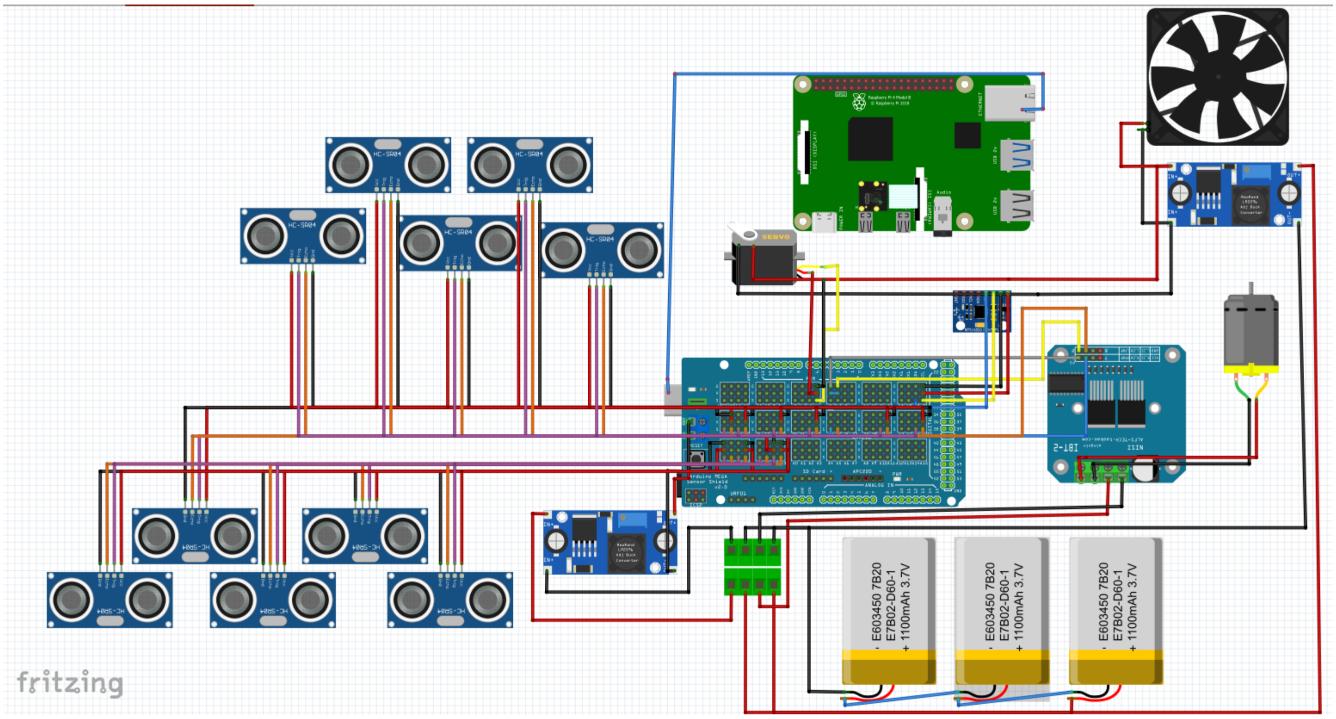


Figure 26: Vehicle electronic - electrical scheme

2.6 WIRING DIAGRAM

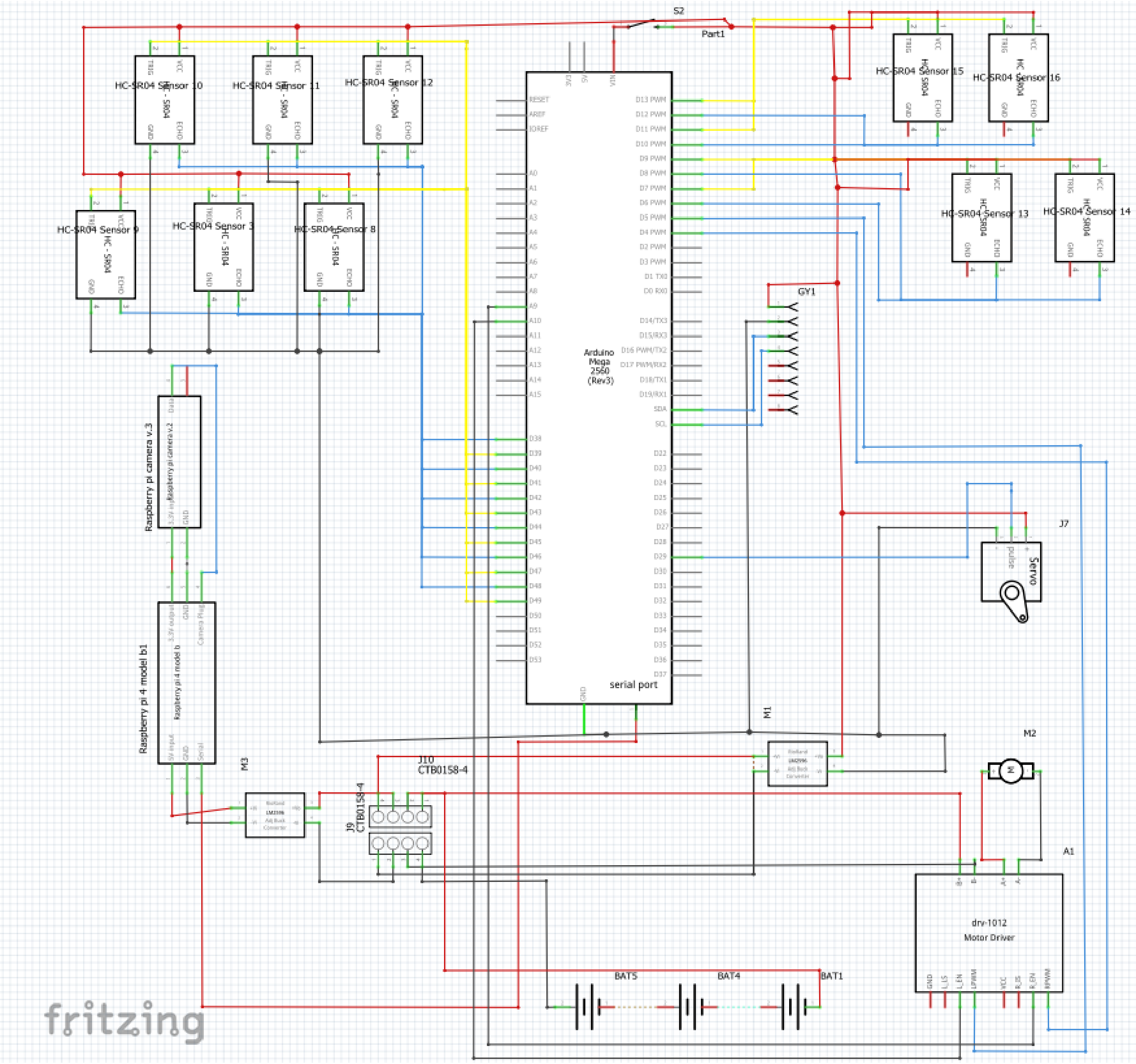


Figure 27: Vehicle electronics: Wiring Diagram

2.7 Bill of Materials

2.7.1 Assembly List

Label	Part Type	Properties
A1	drv-1012	variant variant 1
BAT1	Battery Holders	variant 1100mAh; package lipo-1100mAh
BAT2	Battery Holders	variant 1100mAh; package lipo-1100mAh
BAT3	Battery Holders	variant 1100mAh; package lipo-1100mAh
F1	Fan - Ventilador	variant variant 1; package THT; pins 2; pin spacing 0.13
GY1	MPU-6050 Board GY-521	row double; form (female); variant variant 1; package
HC-SR04 Sensor 10	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 11	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 12	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 3	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 8	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 9	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 13	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 14	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 15	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
HC-SR04 Sensor 16	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
IC1	IC	variant variant 6; package DIP (Dual Inline) [THT]; chip IC
J7	Basic Servo	
J9	Camdembesc CTB0158-4	variant 90° 4 connector; package THT; pins 4; pin spacing
J10	Camdembesc CTB0158-4	variant 90° 4 connector; package THT; pins 4; pin spacing
M1	RioRand LM2596	variant variant 2; chip label LM2596
M2	DC Motor	
M3	RioRand LM2596	variant variant 2; chip label LM2596
Part1	Arduino Mega 2560 (Rev3)	type Arduino MEGA 2560 (Rev3)
Part4	Arduino Mega Sensor Shield v2.0	variant Mega Shield v2.0; pins 325
Raspberry Pi 1	Raspberry Pi 4	revision RPI-4-b; variant Raspberry Pi 4 B; processor Br

Table 3: Assembly List

2.7.2 Shopping List

Amount	Part Type	Properties
1	drv-1012	variant variant 1
3	Battery Holders	variant 1100mAh; package lipo-1100mAh
1	Fan - Ventilador	variant variant 1; package THT; pins 2; pin spacing 0.137in (3.5mm)
1	MPU-6050 Board GY-521	row double; form (female); variant variant 1; package THT; pins 8; pin spacing 0.1in (2.54mm); hole size 1.0mm,0.508mm
10	HC-SR04 Ultrasonic Distance Sensor	variant variant 1; chip LM324
1	IC	variant variant 6; package DIP (Dual Inline) [THT]; chip label IC; editable pin labels false; pins 8; pin spacing 300mil; hole size 1.0mm,0.508mm
1	Basic Servo	
2	Camdenboss CTB0158-4	variant 90° 4 connector; package THT; pins 4; pin spacing 0.2in (5.08mm); hole size 2.7mm; part # CTB0158-4
2	RioRand LM2596	variant variant 2; chip label lm2596
1	DC Motor	
1	Arduino Mega 2560 (Rev3)	type Arduino MEGA 2560 (Rev3)
1	Arduino Mega Sensor Shield v2.0	variant Mega Shield v2.0; pins 325
1	Raspberry Pi 4	revision RPI-4-b; variant Raspberry Pi 4 B; processor Broadcom BCM2837 64-bit ARMv8; part # RPI-4-b

Table 4: Shopping List

2.7.3 Notes

- All resistors are 1/4 watt, 5% tolerance unless otherwise specified.
- Capacitors are electrolytic unless otherwise specified.
- Ensure proper orientation of polarized components.
- Verify all component values and ratings before soldering.

3. Obstacle Management

3.1 OPEN CHALLENGE ALGORITHM

With this challenge, we need to complete three laps while avoiding the wall and trying to do it in the fastest time. This means designing an algorithm ensuring exact navigation and ensuring quick, smooth movement between the sections. The basic idea can be expressed by dividing the section into two zones. This approach makes it possible to solve each problem independently and in turn. The first Zone, called the Safe Zone, contains the majority of the robot's path within it (Blue Zone), as illustrated in the figure below. The second area, which is called the Danger Zone, is the danger zone (10 cm) in which the robot becomes very close to the wall, as illustrated in the figure below (Red zone).

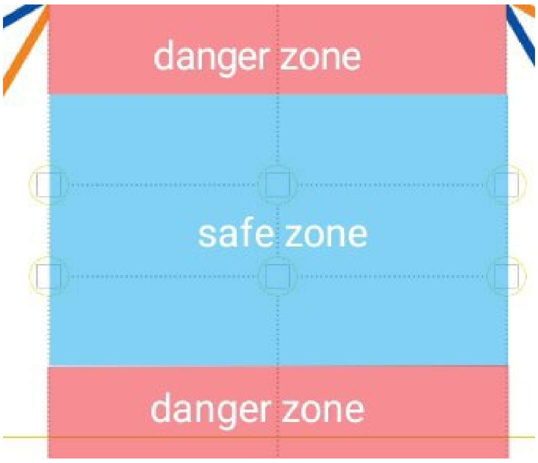


Figure 28: zones Division

3.1.1 Moving at safe zone:

With the Safe Zone, the robot moves safely without the risk of being hit on the wall; hence, with this fact, it is considered good for a robot to keep moving in a straight line and parallel to the wall. The PD controller holds the angle of the robot constant. The gyroscope reading comes in as input to the algorithm, and the setpoint comes after the turn is taken. The formula for error and the angle it requires the servo to change to are as follows from the equation below:

$$\text{PIDOutput} = K_{p(\text{protected})} \times \text{Error}_t + \frac{\text{Error}_t - \text{Error}_{t-1}}{\text{IntervalLength}} \times K_{d(\text{Protected})} \tag{1}$$

$$\text{Error}_t = \text{GyroscopeReading} - \text{SetPoint} \tag{2}$$

3.1.2 Moving into the Danger Zone:

When an Danger zone is entered, movement is unsafe. The robot must move immediately away from the Danger zone until reaching the Safe zone. We used a P controller (just for moving away from the wall) with a relatively high proportional gain:

$$\text{ServoAngle} = K_{p(\text{unprotected})} \times (D_{\text{vertical}} - D_{\text{unprotected}}) \tag{3}$$

$$D_{\text{vertical}} = \cos(\text{Angle}) \times \frac{D_{S1} + D_{S2}}{S2} \tag{4}$$

Where:

- *ServoAngle* is the turning angle of the servo motor.
- $K_{p(\textit{unprotected})}$ is the proportional gain in the unprotected area.
- $D_{\textit{vertical}}$ is the vertical distance between the robot and the wall.
- $D_{\textit{unprotected}}$ is the unprotected area width.

The vertical distance was chosen to inlcude in the formula because it measures the accurate position of the robot relative to the wall, this yield more accureate results compared to the measured distance alone by one sensor or the average of the two. The distance can be calculated using simple geometry.

3.1.3 Detecting turns and direction:

At a distance > 100 cm, the robot detects a turn. However, the robot can be positioned so that it does not allow for parallelism with the wall, providing wrong ultrasonic readings. This is the case of the slight "this rein," where the front wall is detected by the robot using the front ultrasonic sensor, and in case it is impossible to turn safely, the robot moves backward until it reaches a point where the turn is safe. In the first turn, the robot will decide the direction of movement (CW/CCW); this can be done by determining which side of the robot the turn was detected.

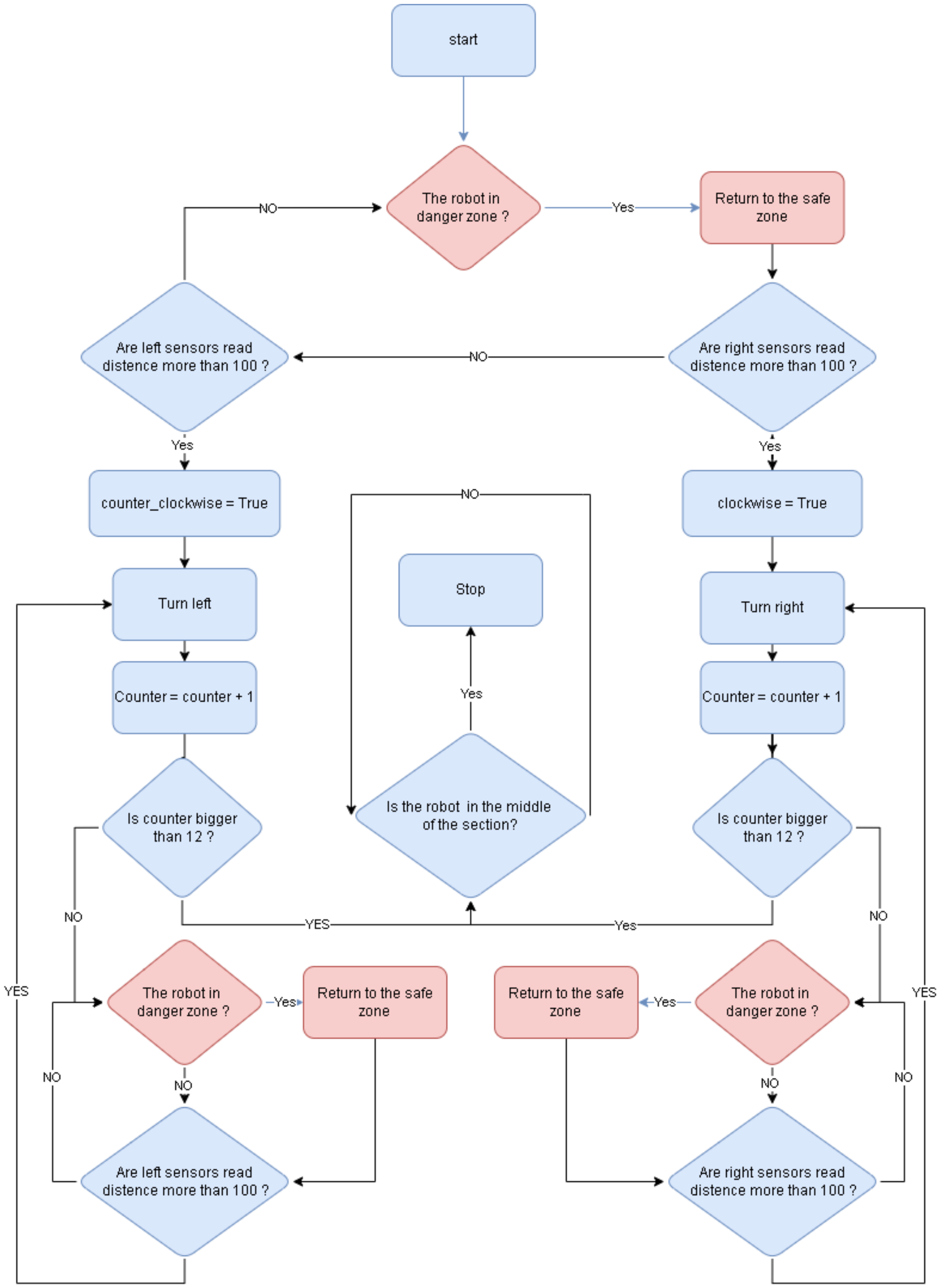


Figure 29: Open challenge flowchart

3.2 COMPUTER VISION:

Computer vision is a cornerstone technology in robotics and automation. This is particularly evident in self-driving cars, where enabling the vehicle to "see" is necessary for detecting and interpreting its surroundings. The robotic system in our task can collect important information to make informed decisions and trigger immediate actions thanks to computer vision (CV) techniques. Identifying the pink parking lot and the red and green pillars is vital to avoid collisions and ensure the robot's path remains clear.

There are several color detection methods available, including HSL (Hue, Saturation, Lightness), HSB (Hue, Saturation, Brightness), RGB (Red, Green, Blue), and HSV (Hue, Saturation, Value). We have chosen the HSV color space because it allows color information (hue) to be separated from luminance, making it work well under different lighting conditions and more robustly.

When working with computer vision, particularly in applications like color detection in videos, it's essential to understand different color spaces and why some are preferred over others. Two commonly used color spaces are HSV (Hue, Saturation, Value) and RGB (Red, Green, Blue).

RGB is a common color space for digital imaging, based on the primary colors of light: red, green, and blue. An RGB image consists of three colors. Each color is a mix of these three components. Fundamental features of RGB include:

- *Red, Green, Blue Channels:*Each pixel's color is determined by the intensity of these three colors.
- *Additive Color Model:* Colors are created by adding light of the three primary colors. Mixing all three lights at full intensity forms white; shutting down these colors creates black.
- *Device-Dependent:* RGB values provide a unique representation of the color, closely tied to just one display device, so consistency is not guaranteed across different applications.

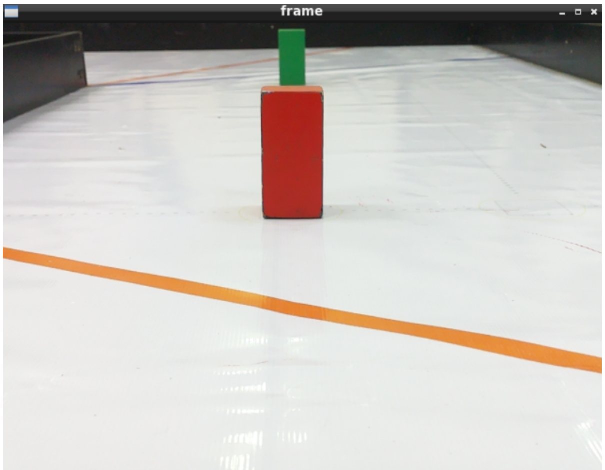
HSV (Hue, Saturation, Value) is used for color analysis more often compared to RGB because it aligns more closely with human perception of colors:

- *Hue:* Represents the type of color, measured along a hue color wheel as an angle in degrees between 0 and 360, where red = 0, green = 120, blue = 240.
- *Saturation:* Measures how deep or vibrant the color is on a scale from 0 (gray) to 100 (full color).
- *Value:* Defines how bright the color can be, with 0 as black (no brightness) and 100 as fully bright.

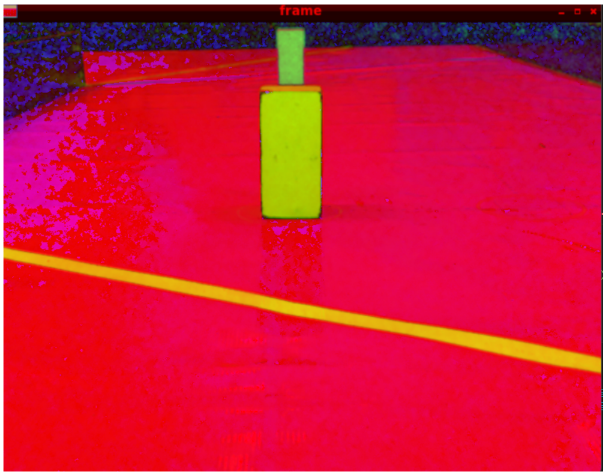
Why Use HSV Instead of RGB for Color Detection?

- *Decomposition of Color and Luminance:* The HSV color space is a transformation of RGB, separating hue from saturation and value. Hue is less impacted by lighting changes, resulting in more accurate detection of relevant regions. In contrast, the RGB color space blends intensity with colors, complicating how different hues appear under changing light.

- *Improved Color Filtering:* HSV filters colors by human perception. For instance, if you are looking for certain colors, increasing the range of hues is a more intuitive and effective approach than adjusting RGB values. In HSV, isolating a single hue is easy, but in RGB, colors can look very different based on brightness.
- *Lighting Robustness:*HSV is more robust to variations in lighting. Changes in light intensity (value) affect hue less than value, making it easier to detect differences independently of slight variations. In RGB, changes in lighting result in concurrent changes across all three channels, making it difficult to adjust while maintaining color matching.



RGB photo



HSV photo

Figure 30: RGB to HSV

3.3 IMAGE PROCESSING:

3.3.1 Setting the Camera Properties:

The shutter speed is adjusted according to the lighting conditions. Shutter speed denotes the duration for which the camera's shutter remains open, allowing light to reach the camera sensor. In high-brightness environments, we reduce the shutter speed to limit the amount of light hitting the sensor when calibrating the HSV value. Conversely, in low-brightness settings, we increase the shutter speed to permit more light to reach the sensor.

3.3.2 Pre Processing:

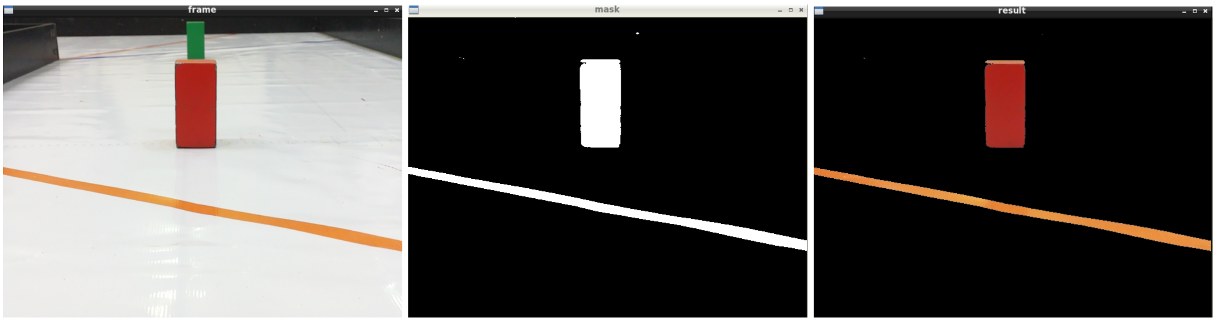
Before isolating the pillars to extract information, it was essential to apply pre-processing to the captured images to eliminate unwanted artifacts, ensuring more accurate and reliable segmentation results. By cropping the frame, we remove distracting elements that might be mistaken for pillars.

3.3.3 Masking:

First, we need to convert the image from the BGR color space to the HSV color space. Then, we identify the upper and lower HSV values of the color we want to isolate(red , green or pink). Next, we examine the value of each pixel in the HSV frame to check if it falls within the desired color range. If the pixel is within the specified range, the corresponding pixel value in the mask will be white; otherwise, it will be black. The resulting mask frame will have white pixels representing the desired color range and black pixels representing all other colors. Our HSV value :

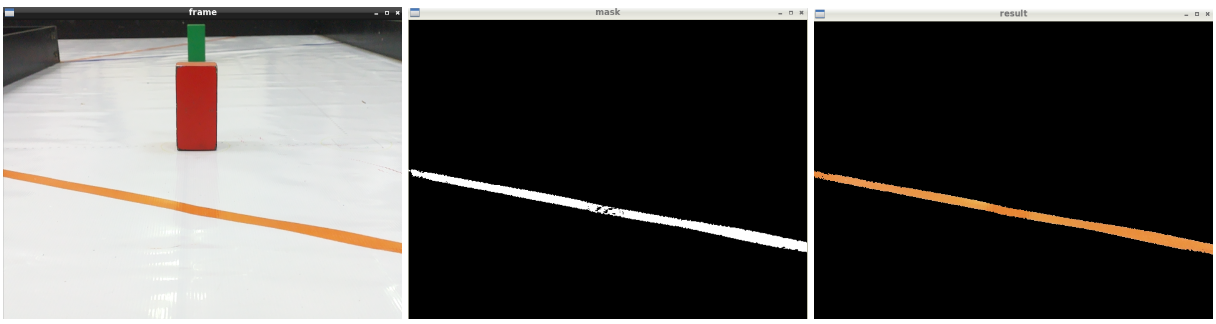
- Red1: Lower value: [100, 43, 130], Upper value: [141, 255, 255]
- Red2: Lower value: [100, 90, 151], Upper value: [111, 255, 255]
- Green: Lower value: [30, 24, 90], Upper value: [76, 213, 255]
- Pink: Lower value:[122,87,181], Upper value:[134,236,255]

Note: this HSV value may change in different lighting condition.



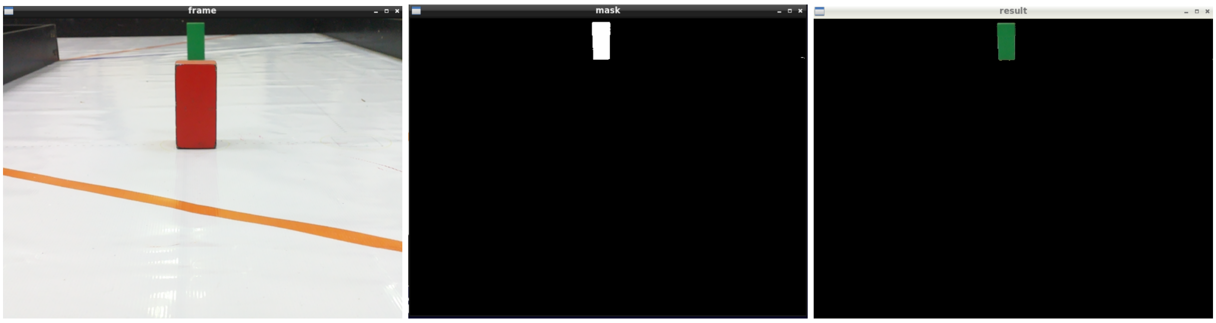
Original Frame Mask Result

Figure 31: Red 1 Mask Analysis



Original Frame Mask Result

Figure 32: Red 2 Mask Analysis



Original Frame | Mask | Result

Figure 33: Green Mask Analysis

3.3.4 Contouring:

Contour detection in computer vision is the process of identifying the boundaries or edges of objects within an image. it is fundamental to many image analysis applications, including image segmentation, object recognition and classification.

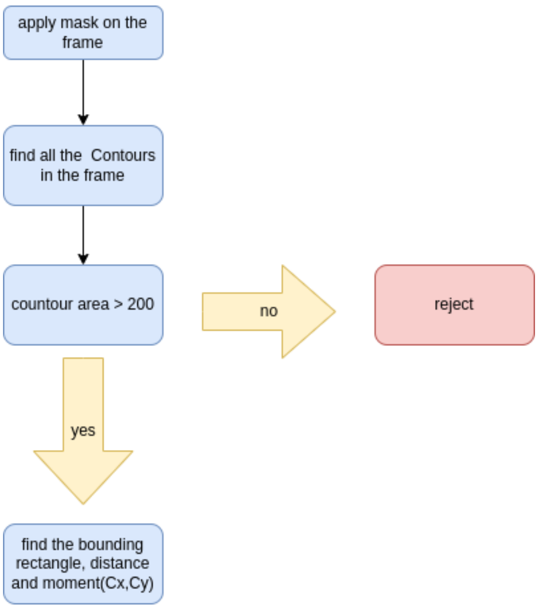


Figure 34: Contouring Process

- *finding the Contour area*

```
cv2.contourArea(contour)
```

- *finding the Bounding rectangle*

```
cv2.boundingRect(contour)
```

- *finding the Moment*

```
cv2.moments(contour)
```

- *finding the Distance* using a formula we created by drawing a curve between the actual distance and the contours area.y

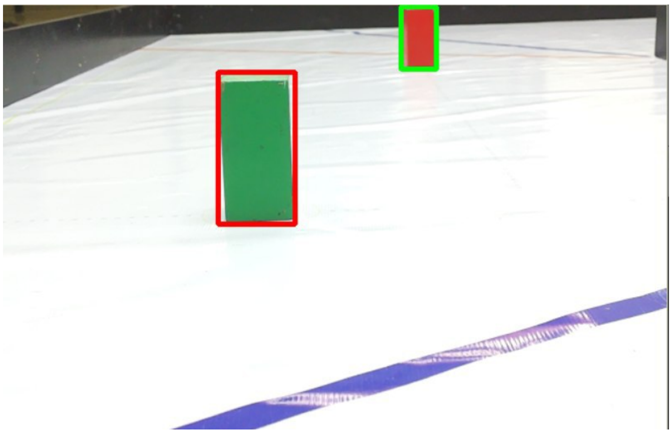


Figure 35: contouring Analysis

3.4 Avoid Pillars Algorithm:

At the start section, the Raspberry Pi captures an image using the camera. It then processes the image and applies a mask to determine if there is any pillar in front of the vehicle. If a pillar is detected and the distance is less than 50 cm, the Raspberry Pi sends a signal to the Arduino ('G' for a green pillar, 'R' for a red pillar, and 'N' for no pillar). The Arduino turns left when it receives 'G', turns right when it receives 'R', or moves forward when it receives 'N'.

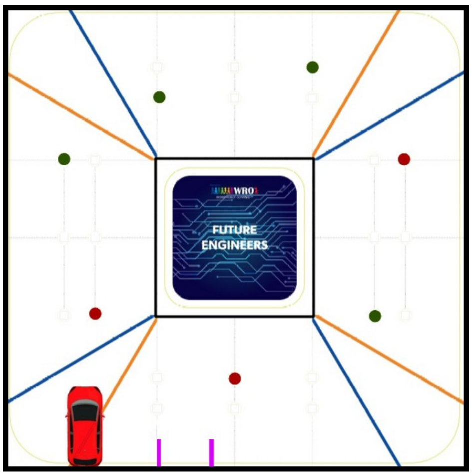


Figure 36: Vehicle corner parking

When the vehicle reaches the corner, the Arduino determines the direction of movement and sends this information to the Raspberry Pi ('L' if the direction is counterclockwise and 'R' if the direction is clockwise). At the end of each section, the Arduino sends 'Hello' message to the Raspberry Pi to take an image, process it, and send the current case to the Arduino, which stores it in a single array consisting of four columns, each on for section, and there is another array that is failed with 1 or 0 depending on the presence or absence of parking lot. The possible cases are GG, G, RR, R, RG, and GR with or without a parking lot.

In the second turn, if there is a red pillar in the third section, The vehicle will make the third turn in the reverse direction of its previous direction.

3.5 Perform of parallel parking:

As mentioned before in the computer vision and image processing section, the robot captures an image at the start of each section to check for a parking lot. The robot determines the condition to be GG or RR based on whether it was rotated clockwise or counterclockwise. Once the robot reaches the parking section, it starts going to the parking lot and park between the parking lot.

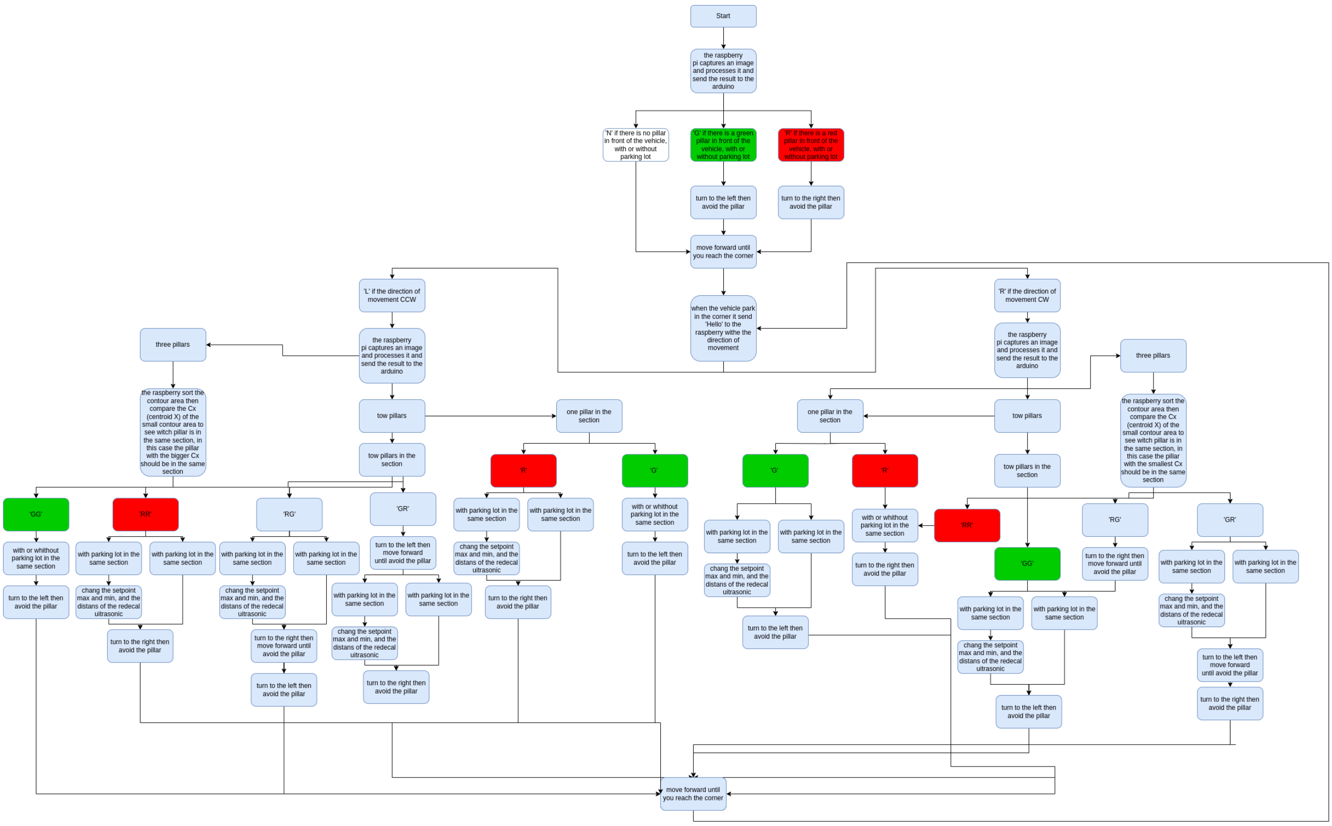


Figure 37: Obstacle challenge flowchart (for high quality image, click here).

3.6 Problems We Encountered:

3.6.1 Wave Interference:

When two sensors operate from the same side at the same time, interference of the sound waves they emit occurs. To solve this problem, we followed the following method, which involves allowing two sensors from different sides to operate each time while stopping the other sensors. This means we take readings from R1 and L2, then R2 and L1, then F and B. This prevents wave interference when measuring distance.

3.6.2 Small Turn Angle and Sliding:

Initially, due to the absence of a differential gear, the turn angle was very small and not smooth, requiring a large distance to reach the desired angle. Additionally, the front tires would slip. These problems were solved by adding a differential gear.

3.6.3 Flickering:

refers to the visible change in brightness or color of the captured frames in a video stream, occurring rapidly and intermittently. When using a Raspberry Pi 4 with a Camera Module 3, flickering can cause the video output to appear unstable and detract from its intended use. We encountered this issue during our project and resolved it by adjusting the shutter speed.

```
picam2.set_controls({"ExposureTime": shutter_speed})
```

4. Suggestions and future modifications

4.1 Open Challenge:

Sometimes, the use of a gyroscope sensor can result in incorrect setpoint calculation. There is evidence for this in that after the robot has turned, it may not be turned precisely 90 degrees, maybe less or more. So, a correction formula is applied that corrects the reading from the gyro and calculates the correct set point.

$$\text{Angle} = \tan^{-1}\left(\frac{D_{S1} - D_{S2}}{D_{S1S2}}\right) \tag{5}$$

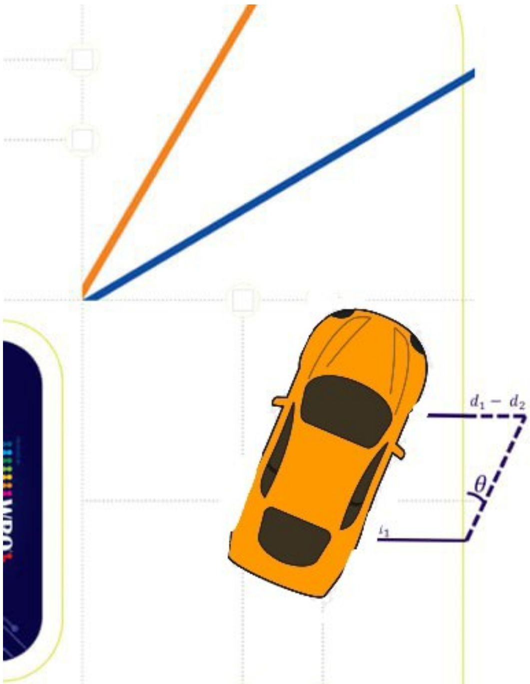


Figure 38: Angle Calculation

Final output can be calculated using the formula below:

$$\text{ServoAngle} = \text{Angle} + \text{PIDOutput} \tag{6}$$

Where:

- *PIDOutput* is the calculated servo angle using PD only.
- *SetPoint* is the reference angle (in which the robot is parallel to the wall).

- $Error_t$ is the difference between the current reading and *SetPoint* in time t.
- K_p is the proportional gain.
- K_d is the derivative gain.
- *IntervalLength* is the time period to take one measurement.
- D_{S1} is the front right/left sensor reading.
- D_{S2} is the rear right/left sensor reading.
- D_{S1S2} is the distance between the front right/left sensor and the rear right/left sensor.
- *Angle* is the angle between the robot and the wall.
- *ServoAngle* is the final servo angle.

4.2 Obstacle Challenge:

There are a few issues with the current vehicle, which utilizes Raspberry Pi Camera Module 3. For this, we suggest upgrading to the Intel RealSense Depth Camera D435 since it will resolve all of these issues and improve reliability in general. The camera is used to provide accurate depth perception so it can measure how far away Pillars and other solid obstacles are. In addition, a wider field of view helps detect obstacles. Its integrated Inertial Measurement Unit (IMU) provides motion tracking better depth accuracy and more stable than ever. This is especially useful in dynamic environments where the vehicle needs to quickly respond.

4.3 Differential:

In our current design, we utilized a LEGO differential; however, we observed some shortcomings, specifically its low durability and the high flexibility of the LEGO axles. To enhance the vehicle's performance, we recommend using a metal differential with metal axles, which offers greater durability and reduced flexibility.

4.4 Micro-controller Upgrade:

In our vehicle, we utilized an Arduino Mega 2560 to control the motors and process sensor outputs, and a Raspberry Pi 4 with a camera to capture images, detect pillars, and send signals to the Arduino for appropriate actions. However, we observed some limitations, particularly in the computational power and integration complexity of using two separate devices. To enhance the system's performance and streamline the architecture, we recommend using an NVIDIA Jetson Nano. The Jetson Nano provides significantly more computational power, allowing for advanced algorithms, real-time data processing, and machine learning applications. Designed specifically for AI applications, it can handle deep learning and image processing tasks, which are essential for complex navigation and obstacle avoidance. This upgrade would consolidate the tasks currently managed by both the Raspberry Pi 4 and the Arduino into a single, more powerful unit.

4.5 Sensor Upgrades

4.5.1 Distance Measurement:

The current setup using Ultrasonic Sensors (HC-SR04) presents several limitations. To address these issues and improve system reliability, we recommend upgrading to Time-of-Flight (ToF) Sensors (VL53L1X) or LiDAR (RPLIDAR A1). This upgrade will resolve the existing problems(ex. Delay in responding) and enhance overall performance. ToF sensors and LiDAR deliver more accurate distance measurements, which are crucial for detecting how far away the surrounding objects are. Furthermore, they offer a faster response time, essential for real-time wall and obstacles detection. Both sensors also provide better range and resolution, significantly boosting navigation capabilities.

4.5.2 MPU Upgrade:

We identified some shortcomings in accuracy and stability with the MPU-6050 used in our current vehicle. To address these issues and improve the system's performance, we suggest upgrading to the Bosch BNO055. The BNO055 features an advanced 9-axis sensor with integrated sensor fusion algorithms, offering enhanced accuracy in orientation and motion tracking. Additionally, it provides better stability by effectively managing drift and noise, ensuring consistent performance in dynamic environments.

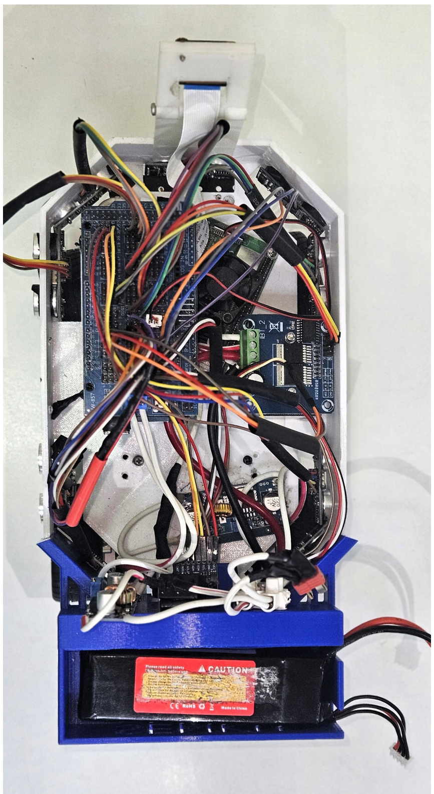
4.6 Power Management System (PMS)

Our current design employs basic power distribution using a Li-Poly battery and DC-DC converters; however, there are opportunities to improve battery management and power stability. To address these issues, we suggest upgrading to a Battery Management System (BMS). A BMS monitors individual cell voltages, temperatures, and overall battery health, which helps extend battery life and ensures safe operation. Additionally, it provides efficient power distribution to all components, maintaining stable voltage levels and preventing power fluctuations.

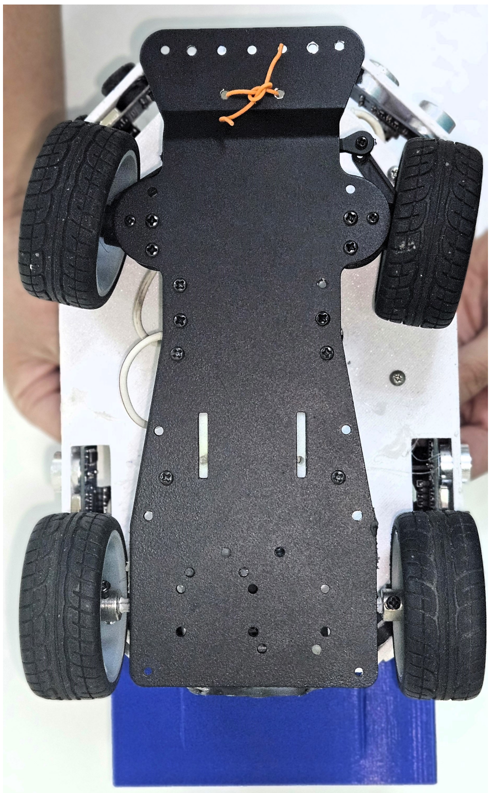
4.7 Conclusion

The improvements will provide better processing power, more accurate sensing, and more reliable control, helping to achieve higher performance in navigating and completing the challenges autonomously
.

5. Vehicle Photos



Top View



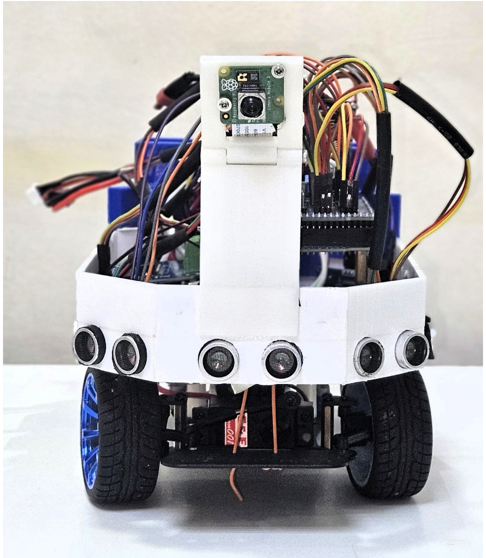
Bottom View

Figure 39: Robot views 1

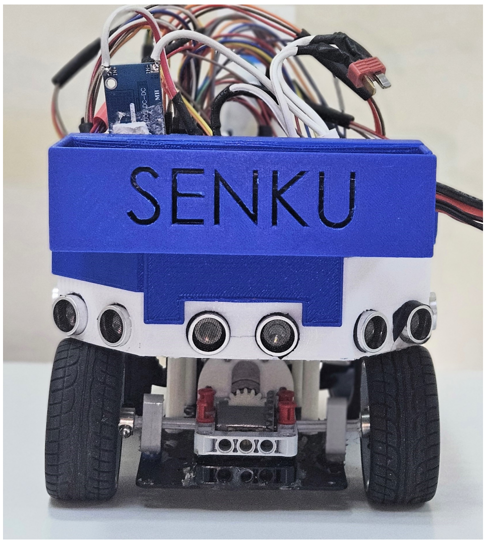
Annotations:

Top View:

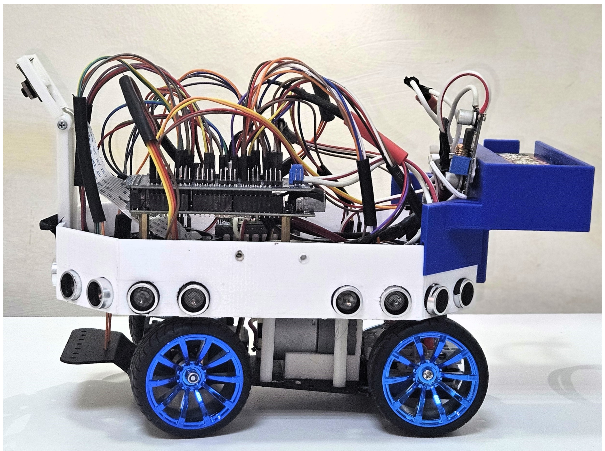
- **Arduino mega:** Main controller for the vehicle.
- **Raspberry pi:** Micro processor for the vehicle.
- **Wiring Layout:** Shows the connections between components.



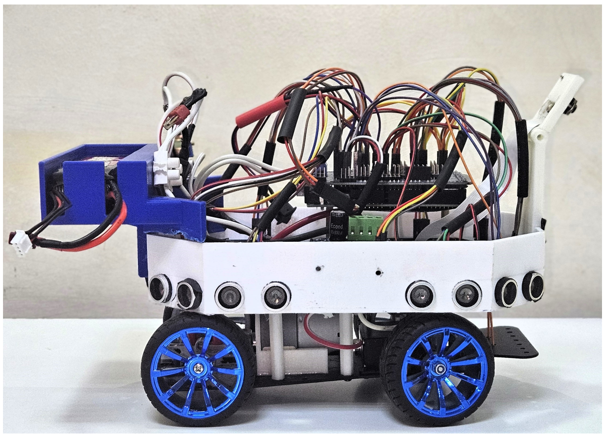
Front View



Back View



Side View (Left)



Side View (Right)

Figure 40: Robot views 2

Annotations:

Front View:

- **Ultrasonic Sensor:** Used for wall detection.
- **Steering Mechanism:** Controlled by the servo motor.

Side View:

- **Battery Pack:** Provides power to all components.
- **Motor Driver:** Controls the DC motor.

6. Team Photo

Figure 41: Team Photo

References

[1] R. N. Jazar, *Advanced Vehicle Dynamics*. Springer, 2019, ISBN: 978-3-030-13062-6. DOI: 10.1007/978-3-030-13062-6. : https://link.springer.com/book/10.1007/978-3-030-13062-6.

[2] S. International, "Motortoy car sample lesson plan," SAE Learn Education, 2016. : https://www.sae.org/binaries/content/assets/cm/content/learn/education/motortoycar-samplelessonplan.pdf.

[3] W. H. Crouse and D. L. Anglin, *Automotive Mechanics*. McGraw-Hill, 2014. : http://opac-library.adamasuniversity.ac.in/cgi-bin/koha/opac-detail.pl?biblionumber=14573.

[4] L. Group. (2023). "Sustainability - product safety - materials," : https://www.lego.com/ms-my/sustainability/product-safety/materials#:~:text=ABS,as%20LEGO%20DUPLO%C2%AE%20bricks.

[5] Arduino, "Arduino mega 2560," : https://store.arduino.cc/products/arduino-mega-2560-rev3.

[6] H. Technology, "Bts7960 motor driver," : https://uk.farnell.com/c/sensors-transducers/sensors/ultrasonic-sensors.

[7] E. Wings, "Mpu 6050," : https://www.electronicwings.com/sensors-modules/mpu6050-gyroscope-accelerometer-temperature-sensor-module.

[8] F. UK, "Ultrasonic sensor," : https://uk.farnell.com/c/sensors-transducers/sensors/ultrasonic-sensors.

[9] S. A. Saleh, D. Jewett, and S. A. R. Panetta, "The analysis, modeling, and capabilities of grounding system designs," *IEEE Transactions on Industry Applications*, vol. 58, no. 5, pp. 5908–5920, 2022.

[10] Besomi, "XL4005 DC-DC adjustable step-down module power supply converter," : https://besomi.com/product/xl4005-dc-dc-adjustable-step-down-module-power-supply-conver

[11] Moviltronics, "Servo MG996R," : https://moviltronics.com/tienda/servo-mg996r/.

[12] Techmaze, "Product 99187969," : https://store.techmaze.ae/product/99187969.

[13] Banggood, "ZOP Power 11.1V 2700mAh 3S 30C Lipo Battery XT60 Plug," : https://www.banggood.com/ZOP-Power-11_1V-2700mAh-3S-30C-Lipo-Battery-XT60-Plug-p-1116169.html.

[14] Modern Physics, "Ackermann Steering Geometry," : https://modern-physics.org/ackermann-steering-geometry/.

[15] MIT, "Differential," : https://web.mit.edu/2.972/www/reports/differential/differential.html.

[16] color spaces : https://pythongeeks.org/color-spaces-and-conversion-in-opencv/.

[17] object tracking : https://docs.opencv.org/3.4/df/d9d/tutorial_py_colorspaces.html.

[18] Raspberry pi 4 product brief: https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-product-brief.pd.

[19] Shutter speed : https://github.com/raspberrypi/picamera2.

[20] Raspberry camera module 3: https://www.raspberrypi.com/documentation/accessories/camera.html.

[21] JSumo 1000 rpm: https://www.jsumo.com/jsumo-titan-dc-gearhead-motor-12v-1000-rpm-hp

[22] contour detection: https://ieeexplore.ieee.org/abstract/document/5459410.

[23] contour detection: https://www2.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-17.pdf.

[24] contours in OpenCV: https://docs.opencv.org/4.x/d3/d05/tutorial_py_table_of_contents_contours.html.

[25] Arduino mega shield: https://www.google.com/url?sa=i&url=https%3A%2F%2Fwww.rajguruelectronics.com%2FProduct%2F1352%2FSensor%2520shield%2520for%2520Arduino%2520Mega.pdf&psig=AOvVaw1U4RU4A_ZUrjizCrHHNAXw&ust=1721493731086000&source=images&cd=vfe&opi=89978449&ved=0CBcQjhxqFwoTCNjk6IzGs4cDFQAAAAAdAAAAABAE.

[26] Wheels: https://nl.aliexpress.com/item/4001168072686.html.

[27] IAEME Publication, *Differential*, https://iaeme.com/MasterAdmin/Journal_uploads/IJMET/VOLUME_8_ISSUE_5/IJMET_08_05_019.pdf.

[28] Roymech, *Useful Tables: Drive Differentials*, https://roymech.org/Useful_Tables/Drive/Differentials.html.

[29] SpeedingParts, *Ackermann Steering*, https://www.speedingparts.eu/i/guides-and-information/chassis/ackermann-steering.html.

[30] James Cole, *Ackerman Steering Mechanism Design Project*, https://sites.nd.edu/james-cole/ackerman-steering-mechanism-design-project/.

[31] IJSER Publication, *Mathematical Model to Design Rack And Pinion Ackerman Steering Geometry*, https://www.ijser.org/researchpaper/Mathematical-Model-to-Design-Rack-And-Pinion-Ackerman-Steering-Geomtery.pdf.

[32] Semantic Scholar, *Ackerman Steering Geometry*, https://pdfs.semanticscholar.org/8d78/832b9cfd0361379699bc34380213244d1bef.pdf.