DESIGN AND IMPLEMENTATION OF VEHICLE COLLISION AVOIDANCE SYSTEM

A Thesis report submitted to

Department of Electromechanical Engineering

College of Engineering

By

1.	Nahom Abebe	ETS 0503/12
2.	Ermias Belay	ETS 0813/12
3.	Fkremariam Eyaya	ETS 0826/12
4.	Habtewold Desta	ETS 0833/12
5.	Kemal Siraj	ETS 0855/12

Presented in Partial Fulfillment of the Requirement Bachelor Degree in Electromechanical Engineering

Mr. Molla Wudye

June 10, 2024

Addis Ababa, Ethiopia

Approval

Addis Ababa Science and Technology University

College of Engineering,

Department of Electromechanical Engineering

The thesis report titled

Design And Simulation of Vehicle Collision Avoidance System

is approved for Bachelor of Science Degree in Electromechanical Engineering.

Advisor Name	Thesis Supervisor	Date

Students Approval

1. Nahom Abebe ……………….
2. Ermias Belay ……………….
3. Fkremariam Eyaya ……………….
4. Habtewold Desta ……………….
5. Kemal Siraj ……………….

Abstract

This study presents the design and simulation of a vehicle collision avoidance system aimed at enhancing vehicle safety by accurately detecting obstacles and preventing potential collisions. The rapid increase in vehicles on roads has led to a higher frequency of traffic collisions, posing substantial risks to road safety despite advancements in automotive technologies. Our motivation of project is the alarming rates of road traffic fatalities and injuries worldwide, this research aims to develop effective collision avoidance systems to mitigate risks associated with human error and enhance overall road safety. The methodology involved integrating various electronic components, including sensors, microcontrollers, and actuators, and implementing algorithms for obstacle detection and collision avoidance. The programming was done using Arduino IDE, and the system's performance was validated through simulations in MATLAB Simulink, utilizing a PID controller to manage speed and braking. The results demonstrated high accuracy in obstacle detection and effective speed control, with rapid response time and efficient execution of avoidance maneuvers

Keywords: *Collision Avoidance, Vehicle Safety, Obstacle Detection, PID Controller, Simulation, Embedded Systems*

Acknowledgment

First and foremost, we would like to thank God for his guidance and blessings throughout the course of this research, providing us with the strength and perseverance to complete this work.

We express our deepest gratitude to our advisor Mr. Molla Wudye, for his invaluable guidance, continuous support, and encouragement. His expertise and insights have been instrumental in shaping the direction and outcome of this project.

We are grateful to the Department of Electromechanical Engineering at Addis Ababa Science and Technology University for providing the necessary resources and facilities to carry out this study. Their support has been crucial in the successful completion of our work.

Finally, we would like to thank our families and friends for their constant support and understanding during this challenging period.

Contents

List of Figures

List of Tables

Acronym’s (Abbreviation)

Abbreviation	Description
ADC	Analog-to-Digital Converter
Arduino IDE	Arduino Integrated Development Environment
CRT	Cathode Ray Tubes
DC	Direct Current
EMI	Electromagnetic Interference
KB	Kilobyte
I2C	Inter-Integrated Circuit
LCD	Liquid crystal display
LDR	Light Dependent Resistor
LED	Light Emitting Diode
MATLAB	MATLAB
MHz	Mega Hertz
PID	Proportional-Integral-Derivative
PLC	Programmable Logic Controller
PWM	Pulse Width Modulation
RPM	Revolutions Per Minute
RTOS	Real-time Operating System
UART	Universal Asynchronous Receiver-Transmitter
SPI	Serial Peripheral Interface

Chapter One

Introduction

1.1 Background of the study

The Vehicle Collision Avoidance System project is initially conceived as a final-year endeavor aimed at developing a technology to assist drivers in averting crashes by issuing timely warnings. Collision avoidance technologies are gaining prominence in U.S. motor vehicles, with automakers increasingly highlighting their potential safety benefits. Nonetheless, the precise impact on crash reductions is still being assessed.

The proliferation of vehicles worldwide is reaching unprecedented levels, significantly affecting daily commuting. The sheer volume of vehicles on the roads has reduced the distance between cars, heightening the risk of collisions. The Road/Lane Departure Warning System is a pivotal crash-avoidance innovation that alerts drivers when they veer out of their lane or off the road, thereby mitigating the risk of run-off-road accidents. It's crucial to note that this system merely informs the driver and does not take control of the vehicle. Its primary function is to prompt the driver to steer back into the lane, particularly on relatively straight roads with a radius of curvature of 500 meters or more and in favorable weather conditions.

It's imperative to consider future iterations of this system, taking into account evolving user experiences. The concept of the vehicle collision avoidance system stemmed from personal experiences, underlining its potential to help individuals evade collisions and promote safer driving habits. The statistics regarding road traffic fatalities underscore the urgent need for such preventive measures, especially considering that vulnerable road users pedestrians, cyclists, and motorcyclists comprise half of those succumbing to road accidents globally. It aims to proactively detect and avoid potential collisions before they occur, thereby significantly reducing the incidence of accidents and saving lives and encompasses a multidisciplinary approach that integrates principles from automotive engineering, sensor technology, computer science, human factors, and transportation policy.

The current global scenario demands heightened attention to road safety, with road traffic injuries ranking as the leading cause of death among young people aged 15–29 years. Unfortunately, 91% of road traffic fatalities occur in low-income and middle-income countries, highlighting the urgent need for comprehensive efforts to address this issue. Road traffic accidents result from a myriad of factors associated with the traffic system, including road users, road environments, and vehicles. Efforts to prevent such accidents have historically been overlooked but are now gaining traction, with evidence demonstrating that concerted action, involving various sectors including health, can yield significant success in reducing the road traffic fatalities.

The importance of collision avoidance systems cannot be overstated. They not only protect lives but also reduce the economic costs associated with accidents. As technology continues to evolve, future trends point towards fully autonomous vehicles equipped with even more advanced collision avoidance capabilities, contributing to a significant reduction in traffic accidents and enhancing overall transportation safety.

In Ethiopia, for instance, the majority of accidents in 20014/15 were attributed to human factors, emphasizing the need for holistic approaches to road safety that addresses all facets of the traffic system. By implementing innovative technologies like the Vehicle Collision Avoidance System and adopting comprehensive strategies, we can work towards a safer and more sustainable future for all road users.

1.2 Problem Statement

The rapid proliferation of vehicles on roads has markedly increased the frequency of traffic collisions, posing significant risks to road safety. Despite advancements in automotive technologies, accidents caused by human error, environmental factors, and mechanical failures continue to result in severe injuries and fatalities globally. Traditional safety measures and driver assistance systems have proven inadequate in effectively preventing collisions, especially in real-time scenarios. The alarming global rates of road traffic fatalities and injuries highlight an urgent need for innovative solutions. This research endeavors to develop a robust vehicle collision avoidance system designed to mitigate risks associated with human error, significantly enhance road safety, and substantially reduce the incidence of traffic accidents.

1.3 Objectives of the Research

1.3.1 General Objective

The main objective of this project is to design and implement a vehicle collision avoidance system.

1.3.2 Specific Objectives

The specific objective of this project is to:

- Develop procedures for obstacle detection and collision avoidance.
- Integrate electronic components such as sensors, microcontrollers, and actuators into the vehicle system.
- Design PID controller.
- Simulations conducted in MATLAB Simulink.
- Analysis the simulation.
- Implement the designed algorithms using Arduino IDE.

1.4 Scope of the Thesis

This thesis covers the design, implementing and simulation of vehicle collision avoidance system. The system integrates sensors, microcontrollers and actuators, with algorithms for obstacle detection and collision avoidance implemented using Arduino IDE. Performance is primarily evaluated through simulations in MATLAB Simulink, focusing on speed and braking management with a PID controller.

1.5 Significance of the Research

This research aims to enhance automotive safety by reducing vehicle collisions through a reliable and responsive collision avoidance system. It contributes to the advancement of vehicle safety technologies and informs the design of more effective systems. The findings support the development of safer vehicles and the reduction of accident rates.

Chapter Two

Literature Review

This literature review offers an overview of advancements in vehicle collision avoidance systems, focusing on speed regulation and automatic braking. Manufacturers have improved accuracy and versatility by integrating various technologies. The review assesses studies on sensor integration, algorithm development, and real-world applications, identifying key factors for system selection and implementation.

Adnan M. Al-Smadi et al. propose a design focused on enhancing vehicle safety by detecting and mitigating potential collisions, particularly at the front or rear. Utilizing ultrasonic sensors, the system continuously monitors surroundings, alerting the driver and taking evasive actions when objects are dangerously close. It analyzes object trajectories to proactively deploy security measures like emergency braking. By integrating advanced sensor technology with intelligent algorithms, the system aims to reduce accident likelihood and severity [1]. However, one drawback of this project is the high cost and complexity of implementation. These systems involve sophisticated technologies and integration processes that can drive up the overall cost of vehicles, making them less accessible to a broader range of consumers. Addressing this drawback requires ongoing research to reduce the costs of sensors and other components.

Mahesh A. Rakhonde et al. present enhancements to an intelligent vehicle system aimed at real-time accident detection and minimizing medical support response times. The system implements multiple units to strengthen accident detection and response capabilities, integrating tire pressure monitoring to prevent accidents and MCU nodes for accident detection. Furthermore, the inclusion of MQ7 enables pollution monitoring for environmental insights [2]. On the downside, the problems observed from this paper included continuous regulation of tire pressure, difficult pollution detection mechanisms and environmental controls. Additionally, to implement the environmental detection additional components are required. The expense of these components adds significantly to the manufacturing cost of the vehicle. Therefore, this project on vehicle collision avoidance system is quite expensive to implement in high density in the real-world application.

P. Ramya et al. aim to tackle driver fatigue and intoxication, major contributors to road accidents, with a technical approach. Their system detects and monitors driver fatigue levels in real-time to proactively prevent accidents. It also detects alcohol intoxication, locking the vehicle's ignition system to prevent drunk driving. Additionally, the system controls the vehicle's direction to avoid potential accidents by incorporating features that detect unsafe distances. In case of an accident, a GSM module promptly sends relevant information to authorities or vehicle owners [3]. Utilizing Raspberry Pi and ultrasonic sensors, the system detects forward movement and controls the vehicle's direction, enhancing safety, especially at high speeds or during lane changes. This comprehensive approach addresses primary accident causes attributed to driver behavior, offering a promising solution for effective accident mitigation while leveraging modern technology for enhanced safety measures and timely response mechanisms.

Aditi Padayar et al. introduce a project targeting the detection of alcohol content in drivers to prevent accidents resulting from drunk driving. Utilizing 8051 family microcontrollers, specifically the 89s52 model, the system employs MQ3 alcohol sensors to detect alcohol concentration in the driver's breath. Analog data from the sensors is converted to digital format using an analog-to-digital converter compatible with the microcontroller. The converted data is then stored and compared to predefined threshold values. If the alcohol concentration exceeds the limit, the program controller disables the ignition system using an electromechanical relay. This mechanism effectively prevents accidents caused by impaired driving, emphasizing proactive measures for road safety through technological intervention [4]. However, this project mainly focused on alcohol detection to prevent vehicle accidents, without taking other factors such as vehicle’s speed and obstacle distance into consideration.

Mubashir Murshed et al. tackle car accidents, citing driver carelessness and excessive speed as major causes. They propose using Internet of Things (IoT) technology to address accident prevention and response challenges. Their system employs distance sensors to monitor the vehicle's proximity to obstacles ahead continuously. Upon detecting a critical distance, the system intervenes by controlling the vehicle's speed and alerting the driver to slow down, reducing collision risks. Additionally, in case of a dangerous accident, the system automatically sends an email notification with vehicle details to the responsible party, enabling swift response and assistance.

This proactive approach utilizes technology for real-time monitoring, intervention, accident prevention, and prompt response, emphasizing road safety enhancement [5]. This project mainly focused on the IoT area and doesn’t consider much of the electromechanical aspects of the system.

Rachia Shettar et al. introduce three fundamental circuits aimed at accident prevention and control. The first circuit employs an accelerometer to detect signs of driver drowsiness, triggering an alarm mechanism via a relay to prevent accidents. The second circuit incorporates an alcohol sensor to identify elevated alcohol levels, disengaging the vehicle's ignition process through the relay to mitigate drunk driving risks. Both circuits include mechanisms to detect brake failure, ensuring vehicle safety [6]. Continuity checks verify if the vehicle owner has been alerted to potential brake issues, preempting accidents resulting from brake failure. This comprehensive approach utilizes technology to address multiple accident factors, emphasizing proactive measures for enhanced road safety.

Serag Mounir Manaa et al. present a comprehensive study on the design and implementation of a vehicle collision avoidance system using modern sensor technologies and microcontrollers. The paper reviews the historical development of collision avoidance technologies, tracing their evolution from the first radar-based system demonstrated by Hughes Research Laboratories in 1995. It highlights significant contributions from automakers and research institutions, emphasizing technological advancements and regulatory milestones. The system incorporates an ARM-based processing unit running a Real-Time Operating System (RTOS) and includes components like a camera for environmental monitoring, sensors for collision detection, and control units for preventive actions. The design methodology relies on Raspberry Pi as the central processing unit, interfacing with sensors and cameras. Software components such as OpenCV for image processing and CMake for software building are integrated to ensure smooth operation. [7]. One drawback of this project is its susceptibility to environmental conditions like heavy rain or fog, which can reduce sensor effectiveness and lead to system failures. The integration of multiple sensors increases complexity, raising maintenance costs and the risk of malfunctions. Drawback of this project reveals the false alarms may occur, causing driver distractions while driving the vehicle, while over-reliance on the vehicle collision avoidance system may lead to driver complacency.

A. S. Jamaludin et al. developed a collision avoidance system using sensors and automated braking. They integrated components like Arduino Nano, ultrasonic sensors, and motor drivers. The system employs a PID control algorithm to regulate vehicle speed based on obstacle distance, ensuring smooth deceleration and stopping. Experimental results show the system effectively stops the vehicle within 0.3 seconds of obstacle detection, validating its real-world potential [8]. However, the paper also transparently discusses the limitations and areas for improvement, offering a balanced perspective on the feasibility of widespread adoption of such systems in the near future. The paper notes limitations such as sensor reliance affected by weather conditions, potentially compromising reliability. Additionally, concerns arise regarding the effectiveness of PID controllers, especially in high-speed scenarios where response time is crucial. Real-world testing in diverse conditions is lacking, with controlled experiments dominating the assessment, potentially overlooking variables impacting reliability and performance in unpredictable scenarios.

Dr. Madhu B K et al. present a low-cost vehicle collision avoidance system to enhance road safety. The system utilizes camera technology for constant environmental monitoring, detecting potential crash scenarios and activating control mechanisms. Hardware setup includes Raspberry Pi 3, USB webcam, and interfacing components, detailed alongside software installation and configuration for Windows and Raspbian OS. The integration and experimental setup illustrate a robust design process. Future enhancements align with automotive safety advancements, showcasing a forward-thinking approach to ongoing development [9].

The project's reliance on weather-sensitive sensors like cameras and lasers could compromise effectiveness in adverse conditions, suggesting the need for additional sensor types for improved reliability. Concerns also arise regarding the system's response time, especially in high-speed scenarios where quick decision-making is crucial. Furthermore, while designed to be low-cost, scalability for widespread adoption may pose economic and logistical challenges, particularly in integration with diverse vehicle models and existing infrastructure.

In summary, the literature review on vehicle collision avoidance systems highlights a field marked by ongoing advancements in sensor technologies, algorithm selection, and control strategies. As a crucial tool in ensuring traffic safety, vehicle collision avoidance systems are increasingly deployed worldwide, poised for further growth and development alongside technological advancements.

Chapter Three
System Modeling/Methodology

3.1 Methodology

This research aims to design, develop, and evaluate an object-detecting vehicle collision avoidance system to enhance road safety through timely detection and speed control. The study begins with analyzing the need for advanced collision detection systems, focusing on electric vehicles in various environments. Data from integrated sensors will drive the control system based on programmed distance-speed relations. The research identifies independent variables (sensor types, algorithm parameters) and dependent variables (detection accuracy, response time), operationalized with precise measures for quantitative analysis. Experimental techniques include software simulation and prototype testing. Performance will be evaluated by manipulating vehicle distance to observe speed effects. Data analysis involves statistical techniques, computer algorithms, and qualitative methods, using mathematical models for simulation with a DC motor controlled by a PID controller. Key software tools are Matlab Simulink for dynamic system modeling and Arduino IDE 1.8.19 for microcontroller programming. The prototype, using an Atmega 320 microcontroller, is programmed with C codes to demonstrate the distance-speed relations.

3.1.1 Experimental Setup and Materials Used

The methodology of the vehicle collision avoidance system involves a combination of sensors, actuators, and control mechanism to detect potential collisions and take proactive measures to avoid or reduce them.

1. **Micro controller**

The micro controller will be considered as the brain of the system in which its main function is to analyze and respond to inputs in real-time. The vehicle collision avoidance system's micro controller will be Atmega 328p. It is chosen for this project due to its compact size, low cost, low power consumption, and capability to control various electronic systems and devices. The microcontroller processes sensor data in real-time to analyze the surrounding environment and assess collision risks. It executes algorithms to calculate distances, detect obstacles, and predict potential collisions based on sensor readings. Its main task will be to receive inputs, process them and controls the actuators to take proper actions by interfacing with sensors, processing data, assessing collision risks, implementing control logic, providing feedback, managing power, and ensuring safety.

Figure 3.1. Atmega328p microcontroller

2. Sensor

Selection of sensor is a crucial element for this project. The sensors are used to detect obstacles or other environmental conditions and convert them into electrical signals that can be processed and interpreted by the controller. Ultrasonic sensors are composed of two main parts, a transmitter and receiver to utilize acoustic energy to detect objects and measure distances. from the sensor to the target objects. This type of sensor emits high-frequency noises that rebound off objects in the surrounding area and return to the controller. This makes it an ideal sensor for precise object detection. This sensor is ideal for this project because other sensors such as LDR sensors might be ineffective for distance measurement because they might be absorbed by black objects. Ultrasonic sensors continuously monitor the vehicle's surroundings in real-time, providing feedback to the collision avoidance system about the presence and location of nearby objects.

Figure 3.2. Ultrasonic sensor

3. Actuator

DC motors serve as crucial actuators in our vehicle collision avoidance system project, converting energy into physical motion. They provide precise control over position, speed, and torque, facilitating dynamic braking to slow down or stop the vehicle when detecting imminent collision threats. By activating DC motors in reverse polarity or applying regenerative braking, the system effectively decelerates the vehicle to prevent collisions. These motors enable not only precise motion control and speed regulation but also essential functions such as braking maneuvers, steering adjustments, and obstacle avoidance. Their versatility and controllability make them indispensable components in developing responsive collision avoidance technologies, enhancing automotive safety by adjusting the vehicle's speed based on inputs from the collision avoidance system and external factors like detected obstacles or traffic conditions.

Figure 3.3. DC motor (Actuator)

4. Connectors

Connectors play a pivotal role in establishing electrical connections between devices, components, or systems. Specifically, jumper wire connectors serve to link points on a circuit board or between different components in electronics and electrical circuits. They efficiently distribute electrical power from a source, such as a battery, to various circuit components and subsystems, ensuring minimal voltage drops and power losses. Overall, connectors offer crucial functionalities including physical connection, electrical conductivity, signal transmission, power distribution, mechanical support, modularity, flexibility, environmental protection, and identification within electronic circuits.

Figure 3.4. Connecting wires

5. Indicators

Indicators are devices or components used to provide visual, audible, or tactile feedback to convey information, status, or warnings in various systems and applications. They play a crucial role in enhancing safety, efficiency, and user experience. Liquid Crystal Displays (LCDs) is utilized in the object detecting vehicle collision avoidance systems to provide important visual feedback, information, and alerts to drivers when the system detects obstacles. It also indicates the speed of the vehicle as well as the distance of the detected obstacle.

Figure 3.5. LCD display

6. DC Motor speed control module

A DC motor speed control module is a circuit or device designed to regulate the speed of a DC motor by adjusting its voltage, current, or pulse-width modulation (PWM) signals. The function of a DC motor speed control module is to provide precise control over the motor's rotational speed, allowing for dynamic adjustments to match specific application requirements. Its main purpose is to dynamically adjusts the speed of the vehicle's propulsion motors based on inputs from the collision detection and avoidance system. When an obstacle is detected or collision risk is assessed, the module receives signals from the controller indicating the severity of the threat and adjusts the motor speed accordingly.

Figure 3.6. Speed control module

7. **Power supply**

Power supplies are essential components in various applications, providing electrical energy which are used to power electronic devices, equipment, and systems. For prototyping the project, 9V battery cells will be used to supply the required electrical energy to the system.

Figure 3.7. Battery

3.1.2 Physical Components

The physical components of a mechanical system are fundamental to its design, functionality, and overall performance. These elements encompass the tangible parts that constitute a system, including their sizes, shapes, materials, and spatial arrangements. Understanding and precisely defining these components and their dimensions is crucial for achieving the desired operational efficiency, durability, and manufacturability of the system.

Physical components of the vehicle collision avoidance system include sensors such as ultrasonic sensors, control units, actuators, and communication modules. Each of these components plays a specific role:

- Sensors: Detect obstacles, vehicles, and pedestrians, providing real-time data about the surrounding environment.
- Control Units: Process sensor data, execute collision avoidance algorithms, and generate commands for actuators.
- Actuators: Implement corrective actions such as speed adjustments and braking.
- Communication Modules: is used to facilitate data exchange between the vehicle collision avoidance system and other vehicle systems, ensuring coordinated responses.

3.1.3 Fabrication and Assembly

Fabrication and assembly are critical stages in translating detailed designs into functional and reliable safety features for vehicle collision avoidance systems. These stages encompass designing, producing, and integrating various components like sensors, microcontrollers, and actuators. Precision manufacturing techniques, quality control measures, and rigorous testing are essential to create a robust system capable of accurate operation under diverse conditions. Careful selection and integration of components ensure seamless functioning within the vehicle's system, ultimately enhancing vehicle safety

Since the vehicle collision avoidance system doesn’t require any fabrication of new components, our task was to select appropriate materials and assemble them. Assembly involves the integration of these selected components into a cohesive system within the vehicle. This process requires careful coordination and attention to details.

Assembly stages of the vehicle collision avoidance system:

Stage 1: Preparation and Planning: Assess the vehicle's design to determine optimal component placement. Prepare assembly instructions, including component and wiring diagrams.

Stage 2: Preparing a printed circuit board: Prepare the PCB to support and connect electronic components using conductive pathways etched from copper sheets on a non-conductive substrate.

Stage 3: Sensor Installation: Mount ultrasonic sensors strategically on the vehicle, considering field of view, range, and coverage. Secure sensors with brackets or mounts, ensuring proper alignment for accurate detection.

Stage 4: Control Unit Integration: Install the Atmega328p control unit in a suitable vehicle location to process sensor data and execute collision avoidance algorithms.

Stage 5: Actuator Placement: Position brake and speed control actuators where they can effectively control vehicle systems. Connect DC motors to control units and vehicle systems, routing hydraulic or electrical lines properly.

Stage 6: Wiring and Connecting: Route wiring to connect the ultrasonic sensors, Atmega328p control unit, and DC motor actuators, ensuring paths prevent system interference. Secure wiring harnesses with clips or clamps to protect from abrasion and environmental damage.

Stage 7: Calibration and Alignment: Calibrate sensors and actuators for accurate detection and response. Align sensor fields of view and adjust actuator mounting angles to optimize system performance.

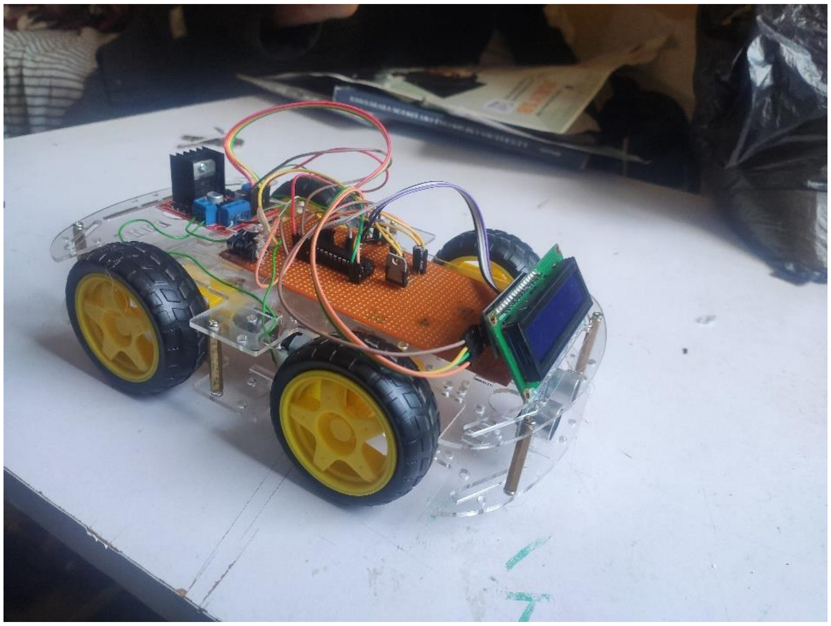


Figure 3.8. Prototype

Challenges and Solutions During Assembly of the Vehicle Collision Avoidance System

The assembly of the vehicle collision avoidance system involved several complex challenges that required careful consideration and proactive measures to ensure the system's reliability and effectiveness. Below are some of the key challenges faced during different assembly stages and their potential solutions:

1. Component Selection and Availability

Challenge: Identifying suitable off-the-shelf components or developing custom components was difficult due to limited availability, long lead times, or compatibility issues. The market often lacks readily available specialized parts required for advanced systems, which can delay the project and increase costs.

Solution: To overcome this challenge, the team adopted a flexible approach by sourcing alternative materials and components that were readily available in the market. This involved extensive research to find equivalent or superior substitutes that met the required specifications without compromising the system's performance. Additionally, the team worked closely with suppliers to expedite orders and reduce lead times.

2. Integration Complexity

Challenge: Integrating the vehicle collision avoidance system components with existing vehicle systems was complex and difficult. The need to ensure compatibility and seamless communication between various electronic and mechanical parts added layers of complexity to the assembly.

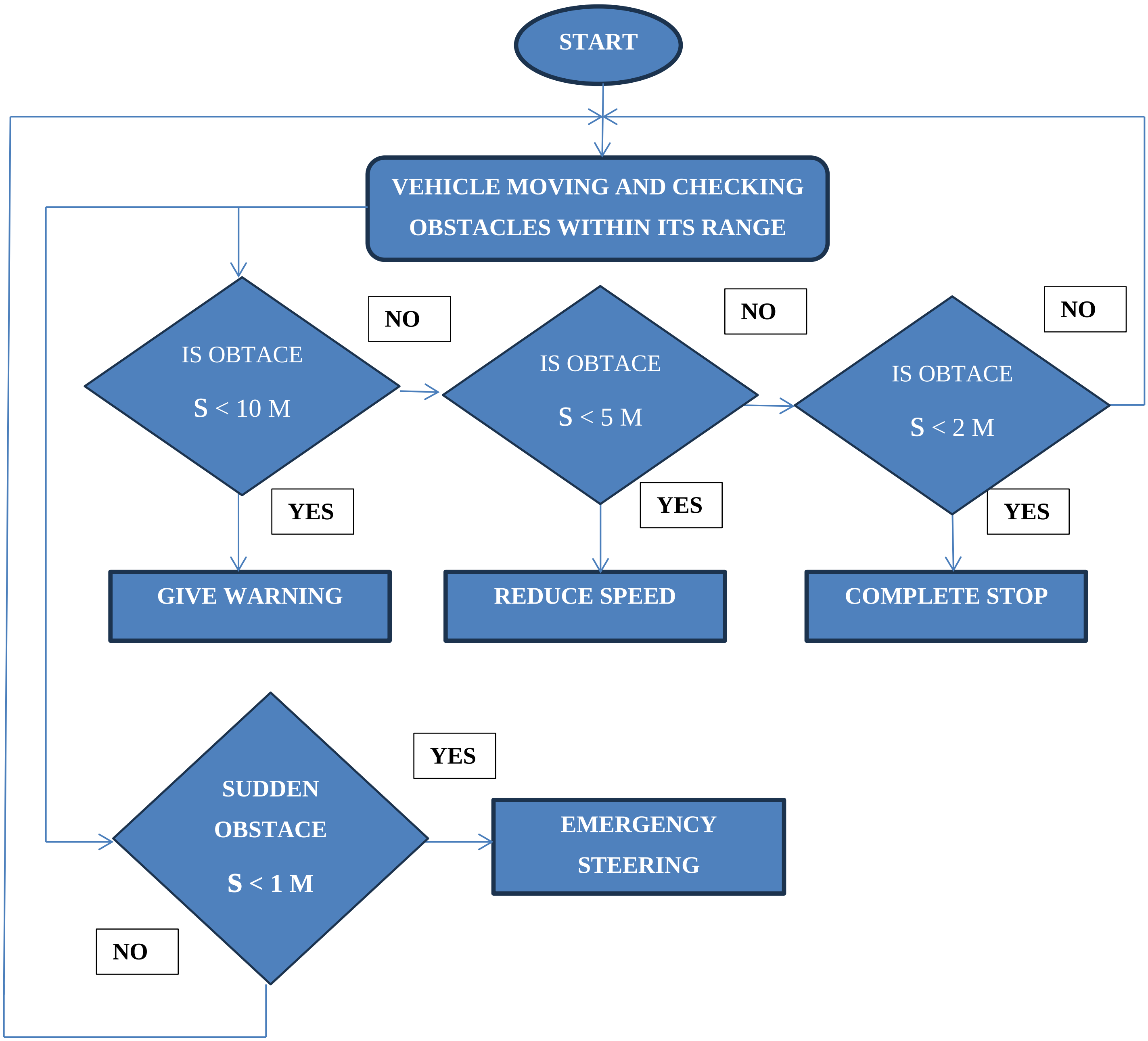
Solution: This issue was resolved by analyzing various assembly techniques and using several reference materials to ensure seamless integration. Furthermore, iterative testing and validation at each stage ensured that all components worked harmoniously, minimizing integration issues and enhancing overall system reliability.

3. Resource and Time Constraints

Challenge: Managing limited resources, time constraints, and budgetary limitations posed significant challenges during the assembly. Balancing the need for high-quality assembly with the constraints of a fixed timeline and budget required careful planning and efficient resource management.

Solution: The team prioritized tasks, optimized assembly processes, and overcame unforeseen obstacles while adhering to project timelines and resource constraints. This approach ensured that the project stayed on track despite the challenges. Collaborative efforts among team members and effective communication with stakeholders facilitated a proactive approach to problem-solving, keeping the project on track despite the challenges.

3.1.4 Operational flow chart



3.2 System Modeling (Mathematical Modeling of Mechatronic System)

3.2.1 DC Motor Modeling

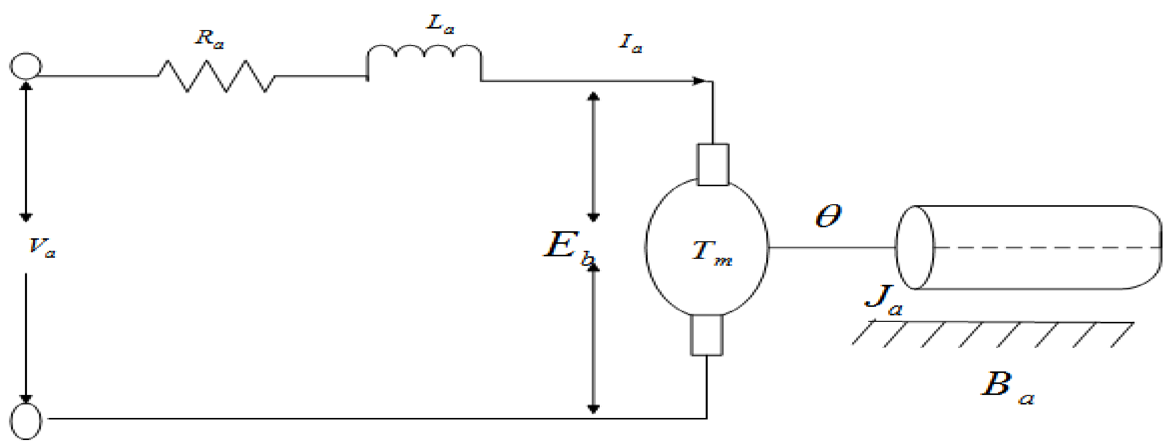


Figure 3.9. DC motor circuit

Where:

V_a is applied voltage to the motor

R_a is internal resistance of motor

L_a is internal inductance

E_b is back emf.

T_m is motor torque

J_a angular inertia

B_a angular friction coefficient

θ angular displacement

T_l is the load torque

I_a is current produced due to V_a.

From above diagram by using Kirchhoff's law:

$$V_a(t) - R_a I_a(t) - L_a \frac{dI_a(t)}{d} - E_b = 0$$

In Laplace domain:

$$V_a(s) - R_a I_a(s) - L_a s I_a(s) - E_b(s) = 0$$

From the above equation:

$$I_a = \frac{V_a(s) - E_b(s)}{R_a - L_a s}$$

Since back emf directly related to motor speed:

$$E_b = K_b\,\omega(s) = K_b s\theta(s)$$

Where K_b is proportionality constant between motor speed back emf. And the equation above becomes,

$$I_a = \frac{V_a - K_b s\theta(s)}{R_a - L_a s}$$

Motor torque T_m :

$$T_m = K_m I_a(s)$$

Where K_m is proportionality constant between motor torque and current. As current increases torque increases.

$$T_m = T_l$$

Where T_l is load torque. And the load torque is equal to the torque applied by the motor.

$$T_l = J_a s^2\theta(s) + bs\theta(s)$$

From the above equation, Since $T_m = T_l$:

$$T_m = J_a s^2\theta(s) + bs\theta(s)$$

From the two equations:

$$K_m I_a(s) = J_a s^2\theta(s) + bs\theta(s)$$

By reordering the above equation:

$$I_a(s) = \frac{J_a s^2\theta(s) + bs\theta(s)}{K_m}$$

By equating the two equations:

$$\frac{J_a s^2\theta(s)+bs\theta(s)}{K_m} = \frac{V_a(s)-K_b s\theta(s)}{R_a-L_a s}$$

$$V_a(s)K_m = [(J_a s^2 + bs)\theta(s)(R_a - L_a s) + K_b K_m s\theta(s)]$$

$$V_a(s)K_m = [(J_a s^2 + bs)(R_a - L_a s) + K_b K_m s]\theta(s)$$

$$V_a(s)K_m = s[(J_a s + b)(R_a - L_a s) + K_b K_m]\theta(s)$$

By rearranging the equation above:

$$\frac{\theta(s)}{V_a(s)} = \frac{K_m}{s[(J_a s+b)(R_a-L_a s)+K_b K_m]}$$

The above equation is written in form of output over input which mean transfer function G(s).

$$G(s) = \frac{\theta(s)}{V_a(s)} = \frac{K_m}{s[(J_a s+b)(R_a-L_a s)+K_b K_m]}$$

$$G(s)= \frac{K_m}{s[(J_a s+b)(R_a-L_a s)+K_b K_m]}$$

This equation shows the relationship between input $V_a(s)$ and output $\theta(s)$.

It relates input voltage with output angular displacement. And the voltage applied to the motor based on the sensor signal.

3.2.2 Vehicle Dynamics

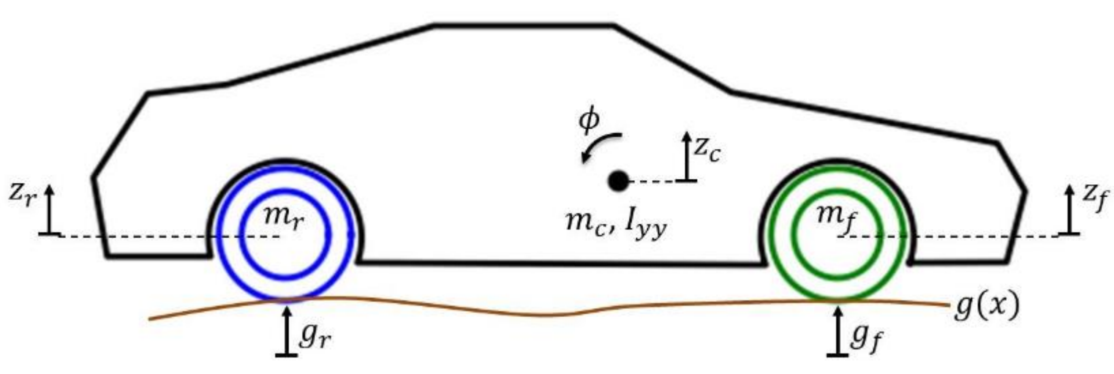


Figure 3.10 Vehicle's free body diagram.

The figure above defines several coordinates and variables:

- Z_c is the vertical displacement of the chassis center of mass (COM) from its initial position
- ϕ is the pitch angle of the chassis,
- Z_f is the vertical displacement of the front wheel COM from its initial position
- Z_r is the vertical displacement of the rear wheel COM from its initial position.
- $g(x)$ is the forcing function (road), and represents the vertical displacement of the road at any given horizontal coordinate x along the road.
- g_f and g_r are the value of this function at the front and rear wheel points of contact, respectively.
- mc, mf, mr, are the masses of the chassis, front wheel, and rear wheel, respectively.
- Iyy is the mass moment of inertia of the chassis about the pitch axis (y axis).

Forces and equations of motion

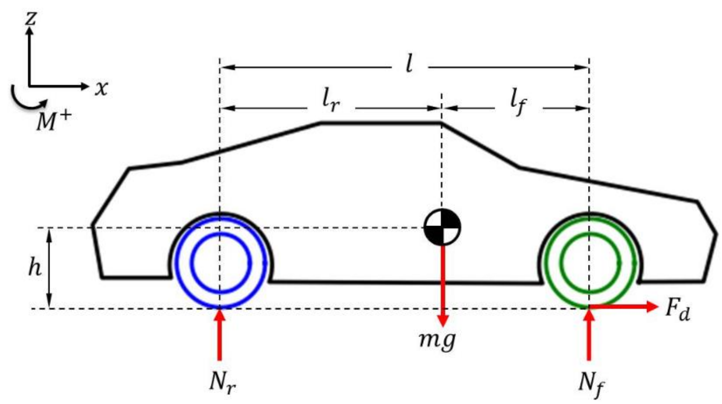


Figure 3.11. Forces acting on the system

$\sum F_x = F_d = ma$

$\sum F_z = N_f + N_r - mg = 0$

$\sum M_{lc} = l_f N_f - l_r N_r + hF_d = 0$

$(l_f + l_r)N_f - l_r mg + hma = 0 \Rightarrow N_f = \frac{lr}{l}mg - \frac{h}{l}ma$

$-(l_f + l_r)N_r - l_f mg + hma = 0 \Rightarrow N_r = \frac{lf}{l}mg - \frac{h}{l}ma$

The distance from the chassis' center of mass and the point where the front suspension attaches to the chassis is where the forces due to the front suspension spring and damper will act on the chassis. Similarly, *lr* is the distance to the rear suspension attachment point. *Fd* is the drive force responsible for accelerating the car.

By considering a negative drive force to be a braking force, even though the car has both front and rear brakes. The drive (or braking) force is related to the car's acceleration by *Fd = ma*, where m is the total mass of the car (*m= mc+mf+mr*) and is the horizontal acceleration of the car.

Note that the system's overall center of mass won't be at exactly the same location as the chassis center of mass, since it also accounts for the center of mass of each wheel, but because the mass of the chassis is significantly larger than the mass of the wheels, it will be relatively close.

The relation between the DC motor modelling and the vehicle dynamics is given by:

$$\mathrm{J_a s^2 \theta(s) + bs\theta(s) = ms^2 x(s)}$$

$$\mathrm{(J_a s^2 + bs)\theta(s) = ms^2 x(s)}$$

$$\frac{x}{\theta(s)} = \frac{ms}{J_a s + b}$$

3.3 Controller Design

In the context of vehicle collision avoidance systems, effective controller design is crucial for ensuring the system's ability to detect potential collisions, evaluate risks, and execute precise maneuvers to avoid accidents. The design process involves developing suitable algorithms and selecting appropriate hardware components that can respond to dynamic and unpredictable real-world environments in real-time.

3.3.1 Selection of Control Strategy

Selecting the appropriate control strategy for a vehicle collision avoidance system is a critical step in ensuring the system's effectiveness and reliability. The control strategy must be capable of processing sensor data in real-time, making rapid decisions, and executing precise maneuvers to avoid collisions. Various control strategies are available, each with its strengths and suitability for different aspects of collision avoidance. For this project the PID controller was selected. PID control is one of the most widely used control strategies due to its simplicity and effectiveness. It involves three components: Proportional (P), Integral (I), and Derivative (D), which are used to minimize the error between a desired setpoint and the actual system output.

3.3.1.1 Controller Objectives of the system

The effectiveness of a vehicle collision avoidance system heavily relies on the controller's ability to respond accurately and promptly to dynamic driving conditions. The primary controller used in the vehicle collision avoidance system is often a Proportional-Integral-Derivative (PID) controller due to its simplicity and robustness. The controller objectives of the vehicle collision avoidance system are as follows:

- Minimum Rise Time: A very short rise time is necessary to quickly react to potential collisions and apply corrective actions, such as braking or speed reduction. Ideal rise time should be less than 0.5 seconds.
- Low Settling Time: Quick settling time ensures that the vehicle reaches a stable and safe state promptly after initiating the avoidance maneuver. Ideal values should be less than 1 second.
- Minimize Overshoot and Oscillations: A critical performance metric of the controller is the minimization of overshoot and oscillations, which can lead to unsafe driving behavior. This includes stable response and damping oscillation to maintain smooth and controlled movements.

3.3.1.2 Comparison of different control strategies

The PID (Proportional-Integral-Derivative) controller offers several advantages over other control mechanisms. Firstly, it provides a balance between responsiveness and stability by incorporating proportional, integral, and derivative actions, allowing it to respond effectively to both steady-state errors and sudden disturbances. This versatility enables PID controllers to achieve faster response times while minimizing overshoot and oscillations, resulting in smoother system operation. Additionally, PID controllers are relatively simple to implement and tune, making them widely applicable across various systems and industries. Their robustness and effectiveness in regulating complex processes, along with their straightforward design and tuning procedures, render PID controllers a preferred choice for many control applications, ranging from industrial processes to robotics and beyond.

3.3.2 Controller Design Method

Controller design in vehicle collision avoidance system involves developing algorithms that can interpret sensor data and generate control signals to the vehicle's actuators (such as brakes, throttle, and steering). The goal is to ensure that the vehicle can maintain a safe distance from other objects, avoid collisions, and provide a comfortable driving experience.

3.3.2.1 Ziegler-Nichols Method for PID Tuning

The values for the PID controllers were gained by using the transient response method (Ziegler-Nichol's) by adjusting the values of Kp, Ki and Kd parameters. Ziegler-Nichols Method is a heuristic method for setting initial gain values based on system response. This method provides a systematic way to determine the proportional (Kp), integral (Ki), and derivative (Kd) gains to achieve desirable system performance.

The steps followed to compute the values of Kp, Ki and Kd parameters are:

- Insert the transfer function of the system (obtained from the dynamic analysis).
- Set P and I gains to zero and increase the P gain until the system oscillates continuously.
- Measure the ultimate gain (Ku) and the ultimate period (Tu) from oscillation.
- Using Ziegler-Nichol's formulas calculate the PID parameters:

P (Proportional): $k_P = 0.6 \cdot k_U$

I (Integral): $k_i = \frac{2k_\rho}{Tu}$ = Kp/Ti

D (Derivative): $k_d = \frac{k_p \cdot Tu}{8}$ = Td∗Kp

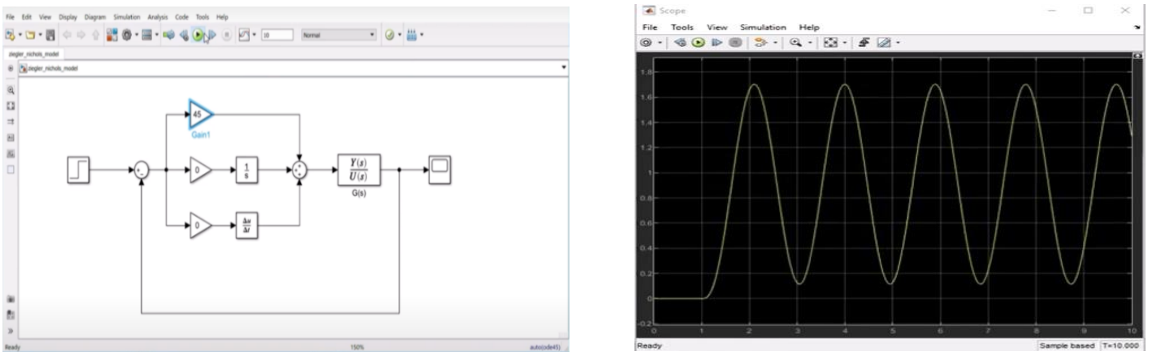


Figure 3.12. Ku and Tu Values

Based on trial-and-error approach, the values of Ku and Tu are obtained from the above simulation. The value of Ku = 0.8 and the value of the period Tu = 0.5. By inserting the values of Ku and Tu in Ziegler-Nichol’s formulas and minor adjustments, we obtained the PID values as:

Table 3.1. Values obtained from calculation

PID Values	
Kp	60
Ki	80
Kd	11.25

3.3.2.2 Stability Analysis

This introduction aims to explore the fundamental aspects of stability analysis in the context of a PID-controlled collision avoidance system simulated in MATLAB. MATLAB, with its robust computational and simulation capabilities, provides an ideal platform for modeling and analyzing such systems. Through this simulation, we can assess the behavior of the PID controller under various conditions, fine-tune its parameters, and ensure that the system operates within safe and stable limits.

In conclusion, stability analysis of a PID-controlled collision avoidance system is a critical step towards developing safer and more reliable autonomous vehicles. By leveraging MATLAB's powerful simulation capabilities, we can rigorously test and validate the system's performance, paving the way for advancements in automotive safety technologies.

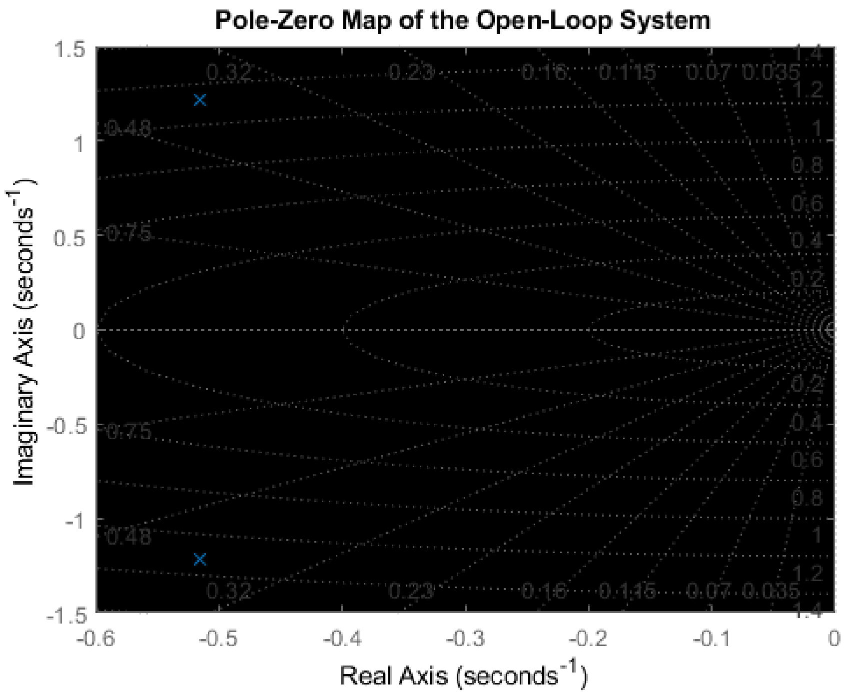


Figure 3.13. Open loop stability

The above figure of pole-zero map of the open-loop system provides insight into the stability of the system before feedback is applied. For the open-loop system to be stable, all poles must be in the left half of the complex plane. Therefore, based on the response above, the system is stable.

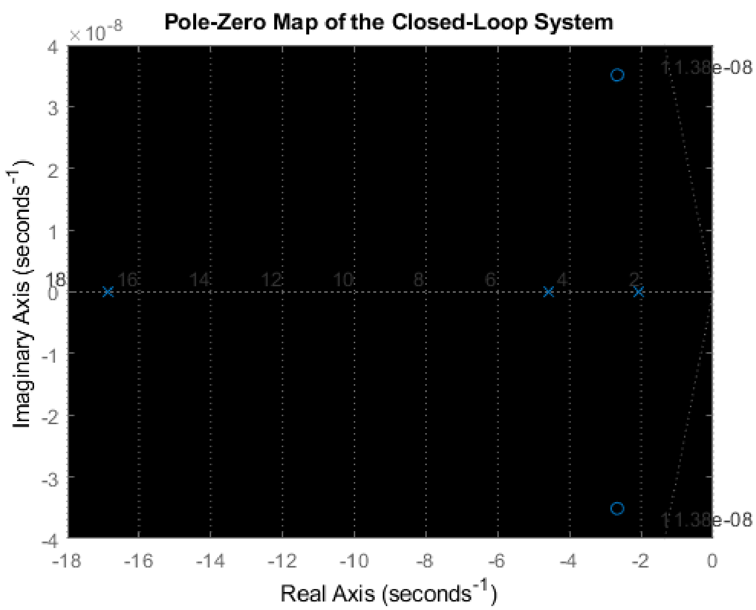


Figure 3.14. Closed loop stability

The pole-zero map of the closed-loop system shows the locations of poles and zeros after feedback is applied. This helps determine the stability of the controlled system. For the closed-loop system to be stable, all poles must also lie in the left half of the complex plane. Therefore, based on the response above, the system is stable.

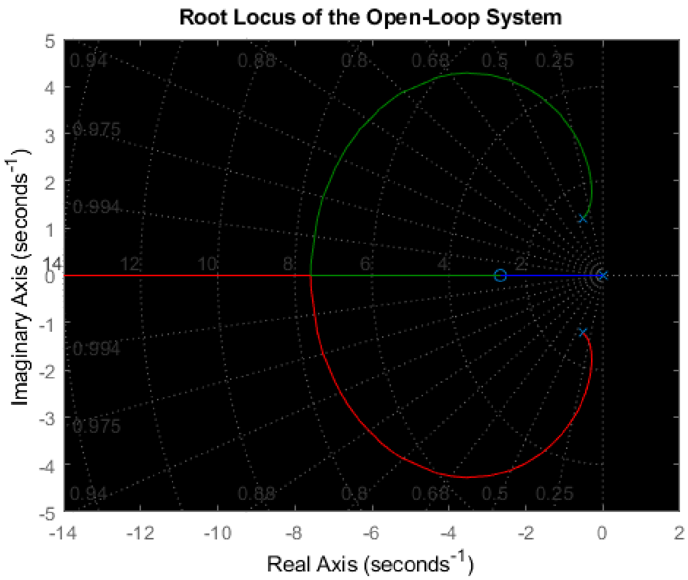


Figure 3.15. Root locus stability analysis

The root locus plot of the open-loop system shows how the poles of the system move in the complex plane as the gain of the controller is varied. This plot is useful for understanding how changes in the PID controller parameters affect the stability and transient response of the system. The above figure represents the root locus which is entirely in the left half-plane for all values of K, thus, the system is stable for all these values of K.

3.4 Electrical and Electronic System Design

Electronic system design is crucial for developing integrated circuits and control algorithms that process sensor data and execute precise collision avoidance maneuvers. Utilizing technologies like ultrasonic sensors and actuators, these systems detect obstacles and take appropriate measures. Real-time data processing identifies potential collision scenarios, enabling proactive interventions. This section outlines the intricate design and implementation of the electrical and electronic systems underpinning the vehicle collision avoidance system based on object detection mechanisms.

3.4.1 Electronic Components

The electronic components of a vehicle collision avoidance system form the technological backbone for detecting potential hazards and initiating preventive actions. These systems rely on a complex array of sensors, processors, actuators, and communication modules to enhance driver safety and prevent accidents.

3.4.1.1 Sensor Selection

Sensor selection is crucial, with ultrasonic sensors chosen for their precision in detecting obstacles. These sensors use acoustic energy, emitting high-frequency noises that rebound off objects to measure distances accurately. Unlike LDR sensors, which can give false readings due to light absorption by black objects, ultrasonic sensors provide reliable, real-time feedback to the collision avoidance system about nearby objects.

Selection criteria

Ultrasonic sensors offer precise distance measurements from a few centimeters to several meters. They are reliable in diverse environments, unaffected by lighting, dust, or smoke, and suitable for harsh industrial settings. Operating on a non-contact principle, they emit sound waves to detect objects, reducing wear and extending lifespan. Easy to integrate with minimal calibration, ultrasonic sensors are effective for vehicle object detection and collision avoidance due to their accuracy, resilience, and cost-effectiveness.

Table 3.2. Sensor Specifications

Ultrasonic Sensor Specifications	
Input voltage (V)	5 DC
Quiescent current (mA)	< 2
Induction angle (°)	< 15
Detection distance (cm)	2 - 450
Accuracy (cm)	up to 0.3

3.4.1.2 Actuator Selection

A critical component of the vehicle collision avoidance system is the DC motor actuator, which translates electronic commands into physical actions like speed adjustments or braking. DC motors provide precise control of position, speed, and torque, essential for steering and braking. They can dynamically brake, reverse polarity, or use regenerative braking to prevent collisions. By adjusting voltage or current, the system controls vehicle speed to maintain safe distances and enhance automotive safety.

Selection criteria

DC motors are ideal for vehicle collision avoidance systems due to their simple control, high starting torque, and adaptability. Speed is easily adjusted by varying the applied voltage, enabling precise speed regulation. Their high starting torque is perfect for applications requiring a strong initial push, such as electric vehicles. DC motors integrate well with control systems like PLCs and microcontrollers, enhancing adaptability in automated systems. They respond quickly to control inputs, allowing rapid acceleration and deceleration, essential for real-time obstacle response. Operating over a wide range of speeds, DC motors ensure smooth and consistent performance, crucial for precise control. Their ease of control, high starting torque, smooth operation, and efficient power utilization make them versatile, reliable, and cost-effective for vehicle collision avoidance systems.

Table 3.3 Motor Specifications

DC Motor Specifications	
Rated Voltage (V)	3~6
Continuous No-Load Current (mA)	150
Min. Operating Speed 3V (RPM)	90RPM
Min. Operating Speed 6V (RPM)	200RPM
Body Dimensions (mm)	70 x 22 x 18

3.4.1.3 Controller Selection

The microcontroller, acting as the system's brain, analyzes and responds to inputs in real-time. For the vehicle collision avoidance system prototype, the Atmega 328p is selected due to its compact size, low cost, low power consumption, and capability to control various electronic systems. It processes sensor data in real-time to analyze the environment and assess collision risks, executing algorithms to calculate distances, detect obstacles, and predict collisions. It interfaces with sensors, processes data, controls actuators, implements control logic, provides feedback, manages power, and ensures safety.

Selection criteria

The ATmega328P's widespread use in Arduino boards makes it highly accessible, supported by a robust ecosystem of development environments, libraries, and community resources. Its ease of programming and extensive libraries enables quick development and adjustments. Supporting multiple communication protocols (UART, I2C, SPI), it interfaces seamlessly with various peripherals and sensors, crucial for interconnected systems like the vehicle collision avoidance system. With an operating frequency of up to 20 MHz, the ATmega328P provides ample processing power for real-time tasks and complex algorithms, making it ideal for the collision avoidance system based on object detection.

Table 3.4. Microcontroller Specifications

ATMEGA328P Microcontroller Specifications	
Program Memory Size (KB)	32
Operation Voltage Max.(V)	5.5
Operation Voltage Min.(V)	1.8
ADC Resolution Max	10
ADC Channels	8

3.4.1.4 Display System

The display system in vehicle collision avoidance systems provides crucial information to drivers, such as driving speed and object distance. Liquid Crystal Displays (LCDs) are commonly used in these systems to convey visual feedback, information, and alerts when obstacles are detected. They indicate both the vehicle's speed and the distance to detected obstacles. LCDs are preferred for their simplicity in configuration and wide availability in the market, making them suitable for this project.

Selection criteria

LCDs are preferred in vehicle collision avoidance technology due to their low power consumption, which is essential for energy-efficient and battery-powered devices. Unlike Cathode Ray Tubes (CRTs) or Light Emitting Diodes (LEDs), LCDs consume less power, making them ideal for portable electronics like our project. Moreover, LCD panels are energy-efficient and generate less heat. They also provide wide viewing angles, ensuring users can view the display from various positions without significant color distortion or loss of image quality. This versatility makes them suitable for large-screen displays and public viewing applications.

Table 3.5. Display Specifications

LCD Specifications	
Display type	Characters
Display range	2-lines X 16-characters
Operation Voltage (V)	5 DC
Module dimension (mm)	80 x 36 x 12
Viewing area size (mm)	64.5 x 16

3.4.2 Software Components

The software components of the vehicle collision avoidance system serve as its digital brains, enabling real-time detection, analysis, and response to potential collision threats. Sophisticated algorithms and programs utilize sensor data to make split-second decisions, ensuring the safety of passengers, pedestrians, and other road users. From advanced signal processing to complex decision-making logic, these components anticipate and mitigate collision risks effectively. Their role is vital in ensuring the successful deployment and effectiveness of the collision avoidance system.

3.4.2.1 Arduino IDE

The software provides a user-friendly interface for writing, compiling, and uploading code to Arduino microcontroller boards. The Arduino IDE supports languages like C and C++, offering features such as syntax highlighting and code completion, along with serial monitoring for debugging. While ideal for prototyping and initial testing of vehicle collision avoidance system control logic, the Arduino IDE may have limitations with more complex tasks like advanced signal processing or real-time performance requirements. In such cases, transitioning to more powerful development environments and languages may be necessary.

Selection criteria

Using the Arduino IDE software was a strategic choice for its seamless sensor integration capabilities. Arduino boards easily interface with common sensors like ultrasonic sensors and DC motors, crucial for the vehicle collision avoidance system prototype. The simplicity of coding in C language, which we're familiar with, streamlined our tasks. Additionally, the IDE's user-friendly interface allows for implementing basic collision avoidance algorithms, such as detecting obstacles and triggering actions like slowing down or stopping the vehicle. Overall, the Arduino IDE provides a versatile and accessible platform for developing vehicle collision avoidance systems, enabling innovation and safer driving experiences through electronic component integration and intelligent algorithms.

3.4.3 System Integration

Integrating electronic components into a vehicle collision avoidance system is a meticulous process, involving selecting, connecting, and programming various elements to ensure seamless operation. System integration entails incorporating sensors, actuators, microcontrollers, and software algorithms to create an efficient and responsive collision avoidance solution.

Sensor integration begins with strategic placement around the vehicle, covering blind spots and critical areas like the front, rear, and sides. Sensors are connected to the vehicle's power supply and communication bus (CAN bus) with shielded cables to minimize interference. Data fusion aggregates and processes sensor data to comprehensively understand the vehicle's surroundings, detecting obstacles on its path.

Microcontroller integration is crucial for data processing, executing collision avoidance algorithms, and controlling actuators. Circuit design integrates the microcontroller into the main control board, interfacing with sensors and actuators. Programs are developed to process sensor data, implement decision-making algorithms, and control actuators.

Actuators, responsible for braking and throttle control, are integrated by connecting them to the power supply. Control signals from the microcontroller are sent to actuators through PWM or analog signals. Feedback loops incorporate sensors within actuators for precise control.

Power management ensures consistent and reliable power supply to all components. Voltage regulators adjust battery voltage for different components. Battery management monitors battery health and manages power distribution to prevent drain or overload.

Software and algorithms integration involves decision-making algorithms for obstacle detection, risk assessment, and action planning. Firmware development codes sensor data processing, decision logic, and actuator control. Real-time Operating System (RTOS) ensures timely task execution and resource management. Software tools simulate system behavior and validate performance before deployment.

Integrating electronic components into a vehicle collision avoidance system demands careful planning and execution. Each component's role is crucial in ensuring the system functions correctly and reliably. By leveraging robust data collection, efficient power management, and advanced algorithms, a cohesive and responsive collision avoidance system can significantly enhance vehicle safety and driver assistance capabilities.

3.4.4 Programming and Control

Programming and controlling electronic components in a vehicle collision avoidance system involves creating software that manages sensor data, processes information, and sends commands to actuators. This process requires a deep understanding of embedded systems, real-time processing, and control algorithms.

Microcontroller programming (ATmega328P), uses ADC to read values from the ultrasonic sensor and implements algorithms to process sensor data and determine the risk of collision. It controls braking force and speed adjustment using PWM signals and uses an LCD to show sensor data and system status.

The programming of this project was done on Arduino IDE (Arduino 1.8.18) software with the standard C programming language. The simulation of the closed loop system was done on Matlab Simulink software for the DC motor control.

Programming and controlling electronic components in a vehicle collision avoidance system involves a combination of hardware interfacing, real-time data processing, and control logic implementation. By using a microcontroller such as the ATmega328P, we can effectively manage sensor data, execute decision-making algorithms, and control actuators to create a responsive and reliable collision avoidance system. This process requires careful planning, robust programming practices, and thorough testing to ensure system reliability and safety.

Chapter Four

Simulation /Experimentation Results and Analysis

4.1 Results

The design and implementation of the vehicle collision avoidance system based on object detection was tested through a series of controlled experiments and simulations designed to evaluate the effectiveness of the system. The results demonstrated a significant accuracy in object detection and speed control of a vehicle. Key performance indicators included the reaction time of the system, the accuracy of obstacle detection, and the success rate in preventing potential collisions with speed reduction and braking. The vehicle collision avoidance system's result and findings are presented in this section.

The Matlab simulation was done by feeding the system's motor a desired speed signal and the output was compared to the input. The system was a closed loop system with a PID controller to reduce any errors in the output. Furthermore, a disturbance signal was added to test the system's ability to reject noise and measure its overall accuracy.

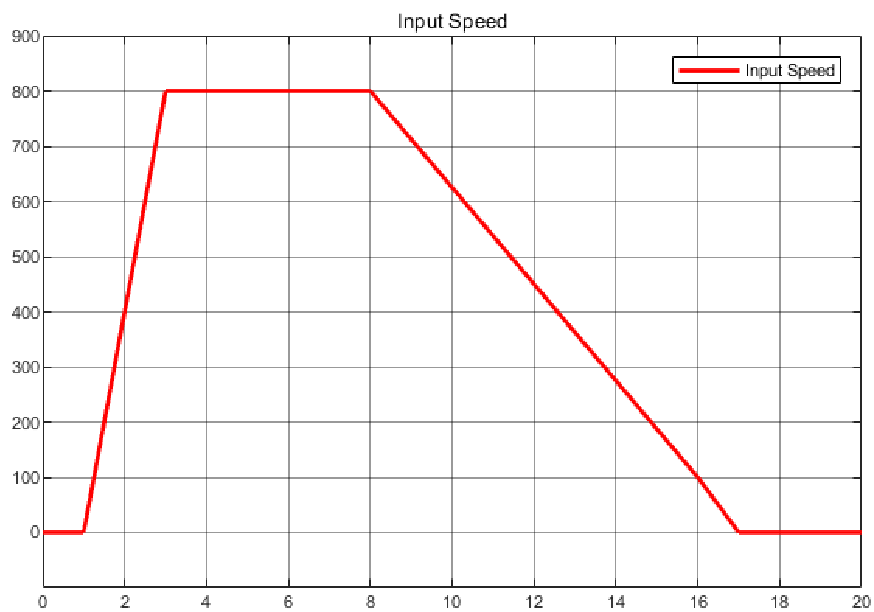


Figure 4.1. Desired speed

The desired (input) speed of the simulation starts from rest and slowly starts to accelerate to 800 rpms and stays at uniform velocity for the next 5 seconds of the simulation. Afterwards assuming the sensors detect obstacles at a certain distance, the vehicle starts to slowly decelerate with time as the vehicle is getting closer to the obstacle. Finally, the vehicle's desired speed will be zero, when the obstacle is within the minimum distance range specified. The simulation, based on this signal block, runs through the system in attempt to get the desired speed as an output.

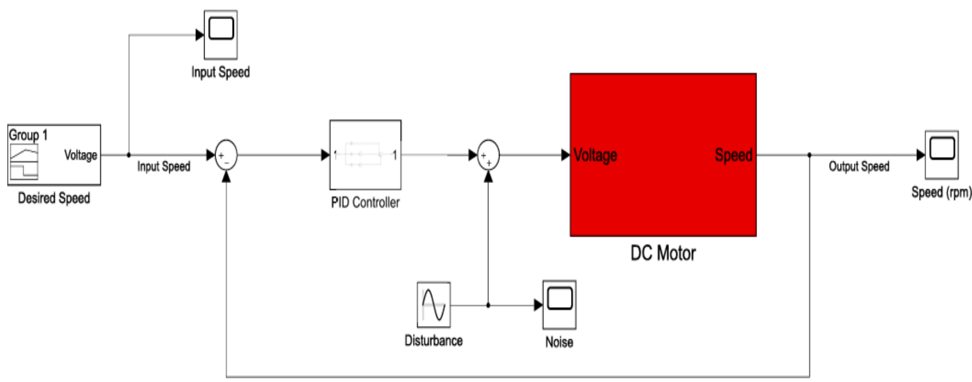


Figure 4.2. Control System

The above figure illustrates the designed control system used for improving the response of the system. The block included a desired speed, scope, a controller, a disturbance, the DC motor block and the output speed block.

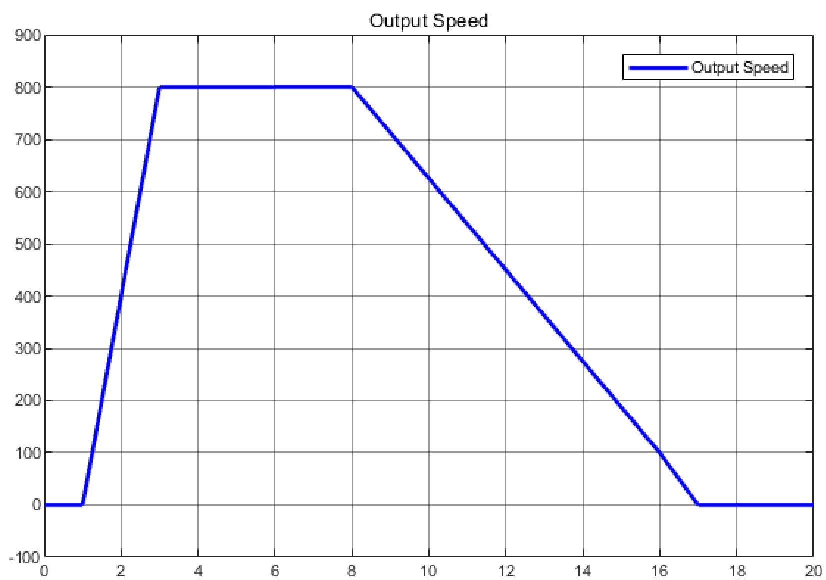


Figure 16. Output Speed

This graph was obtained after the simulation of the desired speed as an input signal through the system. The output speed response of the system was very close to the desired speed. This was obtained by designing an appropriate PID controller to remove any distortions from the output response. This indicates that the input signal from sensors in actual vehicles can be reproduced in the output signal even by avoiding any disturbances. Therefore, with the help of the designed controller the system was able to reject the disturbance signal and provide the desired response with greater accuracy.

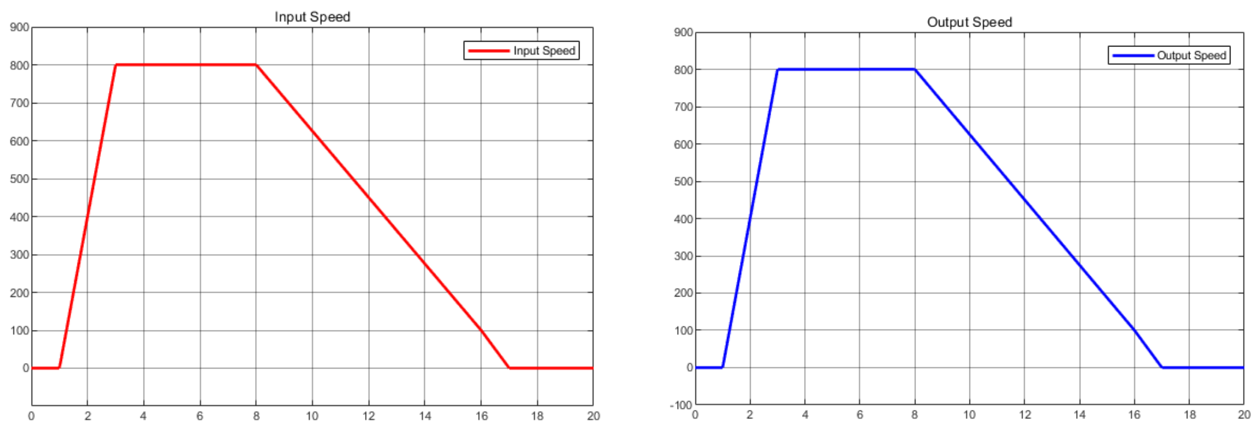


Figure 4.4. Desired Speed vs Output Speed

The above figure compares the findings of the simulation. From the figure we can observe the response of the system and how well the desired output was achieved. This implies that with an appropriate control system design, the motor can be manipulated to produce the desired output.

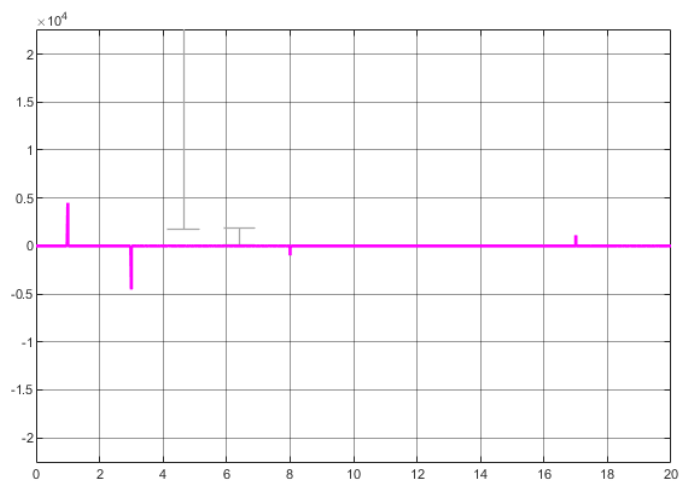


Figure 4.5. Voltage Measure

The above figure shows the voltage fluctuation as the desired speed changes. According to the basic operational characteristics of DC motors, the speed of the motor is directly proportional to the applied voltage, provided the load and other factors remain constant. This relationship allows for a straightforward approach to motor control, where increasing the voltage increases the motor speed, and decreasing the voltage reduces it. Therefore, the results illustrate the impact of voltage control on DC motor.

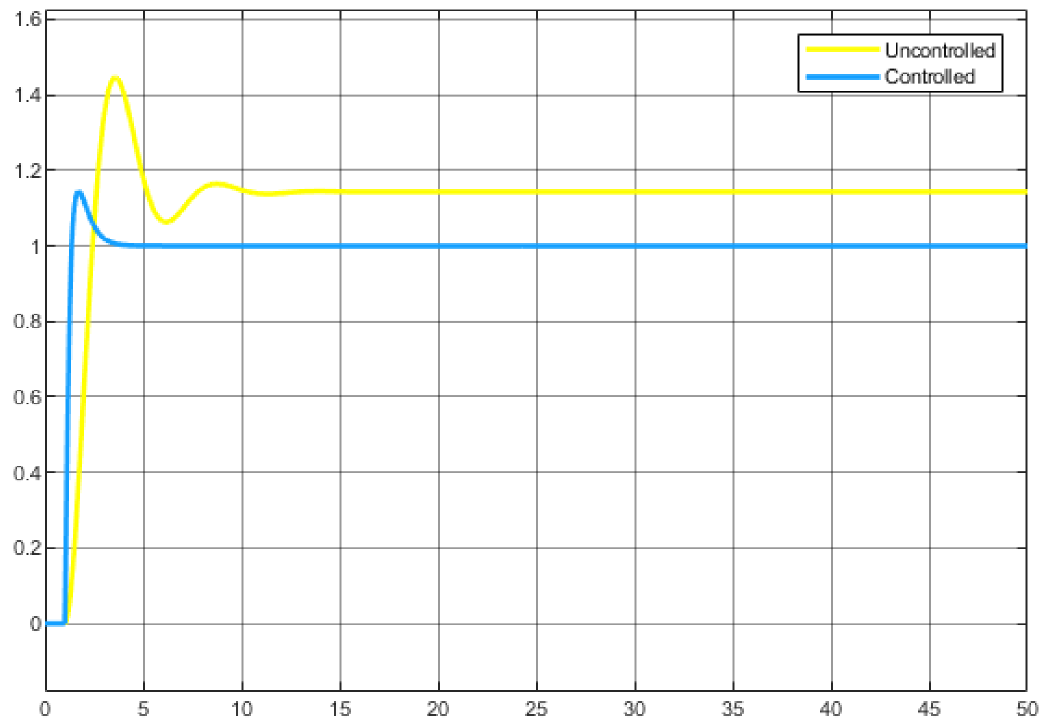


Figure 4.6. Controlled vs Uncontrolled response

The response of the system is indicated in the above figure. We can observe the distortion of the signal response compared to the input for a step input. The system is also affected by the disturbance (sine wave) signal. This negative result had to be modified by designing a suitable PID controller to correct the distortions and provide the desired response. The results show a distorted response which is affected by the disturbance signal. It may lead to unintended consequences or unrealistic behaviors in the simulation. This can obscure the underlying factors influencing system performance and make it challenging to identify potential safety risks or vulnerabilities in the vehicle collision avoidance system design.

Based on the above response the transient state response is obtained as:

Table 4.1. Transient State Response

Transient State Parameters	
Rise Time (seconds)	0.0691
Settling Time (seconds)	0.5079
Overshoot (%)	11.3697

The above table evaluates the performance of the control system by comparing the parameters of the transient response of the vehicle collision avoidance system. The parameters include settling time, rise time, and peak overshoot. The values for the PID controllers were gained by using the transient response Ziegler-Nichols' method of PID tuning by adjusting the values of Kp, Ki and Kd parameters.

4.2 Discussion

The simulation results of the vehicle collision avoidance system based on object detection using MATLAB provided critical insights into the system's performance under a variety of conditions and scenarios. MATLAB, with its powerful computational capabilities and extensive toolboxes for simulation, offered a robust environment for modeling the vehicle collision avoidance system algorithms and testing their efficiency. The simulation results highlighted key aspects of the system's behavior, including its ability to control motors speed as well as reject disturbances that may be present.

The primary focus of the simulation was to evaluate the accuracy and responsiveness of the collision avoidance algorithms. The simulation scenarios included disturbances that are expected to be present in real life scenarios and tested the system's capability to not allow them to interfere with the output response.

The relationship between voltage and speed was also examined in this simulation, and based on the figure we can easily observe the relationship between speed and voltage. Therefore, by applying the direct proportional relationship present between voltage and speed, we can control the vehicle's motor by manipulating the voltage passing through the system.

The MATLAB simulation allowed for the analysis of different control strategies and their impact on vehicle dynamics, providing a clear understanding of how various factors influence the system's performance. Moreover, the selection of the appropriate control strategy is key for obtaining an effective system and ensuring real-time responsiveness of the system to potential collision threats.

One of the drawbacks is the inherent simplification and idealization within the simulation environment. Real-world driving conditions are highly dynamic and unpredictable, involving factors such as varying weather conditions, road surface irregularities, and unpredictable human behaviors, which are challenging to accurately replicate in a simulated environment. Consequently, the system’s performance in the simulation may not fully reflect its effectiveness in actual driving scenarios. Therefore, understanding the mechanisms of voltage control, the associated circuitry, and the impact on motor performance is essential for designing effective motor control systems in real life applications.

4.3 Prototype/Manufacturing Results

The prototype manufacturing results of the vehicle collision avoidance system provide crucial insights into its performance in actual driving environments. These results highlight the system's ability to detect obstacles, make real-time decisions, and execute corrective actions to prevent collisions. The process encompasses rigorous testing under various scenarios, including different road conditions, traffic densities, and weather conditions, to ensure the system's robustness and adaptability. These results not only validate the initial design and simulations but also identify areas for improvement, ensuring that the system meets the highest standards of safety and reliability before market introduction.

4.3.1 Prototype Development

The development of this prototype involves several phases, starting with the design and selection of appropriate sensors and hardware components. Following this, the focus shifts to the development and implementation of the software algorithms that will process sensor data, predict collision scenarios, and determine optimal responses.

Furthermore, rigorous testing and validation in both simulated environments and real-world conditions are crucial to ensure the system's reliability and effectiveness. This verifies the system's capability to be applicable in the real world.

These prototype development stage serves as a vital bridge between theoretical concepts and real-world implementation, allowing us to translate the innovative ideas into tangible solutions that can be tested, evaluated, and refined in controlled environments.

This prototype aims not only to demonstrate the technical feasibility of the vehicle collision avoidance system but also to address practical considerations such as cost, scalability, and integration with existing vehicle systems. By creating a working prototype, the project seeks to bridge the gap between theoretical research and practical application, providing a foundation for further development and eventual commercialization of advanced collision avoidance system based on object detection technologies.

One of the challenges experienced while developing this prototype was the lack of materials available to work with. Some materials were not easily found on the market while others were quite expensive to purchase. Therefore, as a solution to this, we decided to use other alternatives such as cheaper materials and substitutes for the missing components in order to develop this prototype demonstrating the project.

4.3.1.1 Prototype development stages

Stage 1: System Design: includes architecture design of the vehicle collision avoidance system such as the circuit layout, appropriate component selection and algorithm design for the effective implementation of the system.

Stage 2: Material gathering: includes collecting and purchasing the required materials used for the development of the prototype. Materials used include DC motors, ultrasonic sensors, LCD, microcontroller, and other electrical components such as capacitors and voltage regulator.

Stage 3: Hardware Assembly: includes assembling the hardware components, ensuring proper installation and connectivity. This includes mounting sensors on the vehicle, installing the display system, fixing the motor actuators and connecting them to the central processing unit.

Stage 4: Software Development: includes implementing the designed algorithms in the Arduino software. This involves programming the atmega328p microcontroller and developing software for data processing, threat assessment, and control actions.

Stage 5: System Integration: includes integrating the hardware and software components to form a cohesive system. It ensures that the sensors, processing unit, and motor actuators communicate effectively. The software program was transferred to the Atmega328p microcontroller by using FTDI (Future Technology Devices International Limited) device in order to integrate it with the hardware components.

Stage 6: Final Testing and Validation: includes conducting extensive testing to validate the final prototype against all the specified requirements. This includes rigorous field tests in various environments and scenarios as well as placing objects in different distance range to ensure the effectiveness of the speed control of the prototype as well as the braking system when the object is too close to the vehicle.

4.3.2 Prototype Testing and Evaluation

Prototype testing and evaluation represent critical phases in the development of the vehicle collision avoidance systems, providing invaluable insights into the performance, reliability, and effectiveness of the prototypes under real-world conditions. Throughout the testing and evaluation process, we can assess various aspects of the collision avoidance system, including sensor accuracy, algorithm responsiveness, and overall system robustness, to ensure that it meets stringent safety standards and regulatory requirements.

Simulation testing, conducted using software platform MATLAB, allows developers to model and analyze the behavior of the vehicle collision avoidance algorithms in only a virtual environment. Therefore, this stage is crucial for initial validation, as it helps in identifying and correcting algorithmic issues without the risks associated with real-world testing. In order to validate the vehicle collision avoidance system, prototype testing is necessary.

Compared to the simulation, the prototype test results were less rapid in response. However, this was expected as the simulation relies on a virtual environment whereas, the prototype was tested in actual environment. Despite of the this, the prototype test results were beyond our expectations in delivering the desired response. The observed behavior of the prototype indicates the reliability of the system as it was able to perform the intended speed control task with great accuracy and effectively.

The strengths observed from the prototype testing incudes:

- High Detection Accuracy: The prototype demonstrates a strong capability to detect a variety of obstacles, including vehicles, pedestrians, and static objects, using integrated sensors.
- Real-Time Processing: The system’s ability to process data in real-time allows for timely decision-making and execution of avoidance maneuvers. This is crucial for preventing collisions, especially in dynamic and fast-changing environments.
- Algorithm Efficiency: The collision avoidance algorithms are designed to quickly assess risks and determine the optimal response, ensuring that the vehicle can react swiftly to potential hazards.

The limitations observed from the prototype testing incudes:

- Sensor Limitations: The ultrasonic sensors may have limited testing distance, inaccurate readings, and inflexible scanning methods. These limitations can reduce the system's effectiveness in certain environments.
- Real-World Complexity: Factors such as extreme weather, road surface irregularities, and unpredictable human behaviors can impact system performance, potentially leading to false positives or negatives.
- Hardware Reliability: The durability and reliability of the hardware components, especially sensors, need to be ensured for the system to function correctly over time. Environmental factors such as temperature fluctuations, vibrations, and dust can easily affect sensor performance.

Areas for Potential Improvement:

- Enhanced Sensor Fusion: Improving sensor fusion techniques to better handle adverse conditions can enhance detection accuracy. Developing algorithms that can dynamically adjust sensor weighting based on environmental conditions could mitigate sensor-specific limitations.

- Improved Hardware Durability: Developing more robust sensors that can withstand harsh environmental conditions will increase the system's reliability. This could include better weatherproofing, shock absorption, and resistance to temperature extremes.
- User Feedback Integration: Gathering feedback from users during field tests can provide insights into practical usability and acceptance. This feedback should be used to refine the system's user interface and interaction mechanisms.
- Extensive Real-World Testing: Conducting extensive real-world testing in various geographic and environmental conditions will help identify and address limitations. Collaboration with automotive manufacturers and fleet operators for long-term testing can provide valuable data and insights.

4.3.3 Manufacturing Considerations

Manufacturing a vehicle collision avoidance system involves several critical considerations to ensure that the system is not only functional and reliable but also scalable and cost-effective for eventual mass production. Key considerations include component selection, integration, production processes, quality control, and compliance with industry standards. One of the most important considerations is component selection and sourcing. High-Quality sensors selection involves selecting reliable and high-performance sensors.

Another key consideration should be reliability and durability. Designing components to withstand harsh automotive environments, including extreme temperatures, moisture, dust, and vibrations and ensuring that components have a long operational life and can withstand the expected lifecycle of the vehicle is crucial. Additionally, incorporating redundancy for critical components to enhance system reliability and safety is also an important consideration.

The estimated time and cost of producing the vehicle collision avoidance system is hard to predict. Material prices are rapidly increasing on a daily basis and due to advancement in technology newer components could be used to implement the vehicle collision avoidance system in a better way. Furthermore, many new machineries are being invented and production is being automated in many industries. Therefore, estimation of this system's mass production time and cost is difficult to forecast.

By addressing these manufacturing considerations, we can effectively transition the vehicle collision avoidance system prototype from a conceptual model to a commercially viable product. It is crucial to ensure high quality, reliability, and compliance while managing costs to successfully deploy and widely adopt the vehicle collision avoidance system.

4.3.4 Comparison to Simulation Results

The comparison between the prototype of the vehicle collision avoidance system and its simulation results is crucial for validating the system's design and performance. It is a critical step in validating the accuracy and reliability of the simulation models used in the development of vehicle collision avoidance system. This comparison helps in understanding how closely the real-world implementation matches the theoretical models and simulations, and identifies areas where further refinement may be necessary. t serves as a means to validate the accuracy and reliability of the simulation models used during the design phase and assess how well the real-world performance aligns with the expected outcomes. There are different criteria to compare the prototype results with the simulation results.

4.3.4.1 Based on Detection Accuracy

The simulation results yielded high accuracy in detecting obstacles using sensor fusion algorithms and reliable performance in various simulated environments. It also provided consistent detection of static and dynamic objects within the designed range. On the other hand, the prototype result provided in the real-world tests, the prototype generally matched the detection accuracy observed in simulations. The system successfully detected obstacles under controlled conditions and typical driving environments. However, some inconsistencies were noted in complex scenarios, such as heavy traffic or adverse weather conditions, where sensor performance could be affected by external factors.

4.3.4.2 Based on Reaction Time and Responsiveness

The simulation results produced rapid processing of sensor data and timely execution of avoidance maneuvers based on the input signals. Additionally, efficient algorithm performance with minimal latency was observed in the simulated environments. Conversely, the prototype result demonstrated comparable reaction times to the simulation, effectively processing data and executing maneuvers in real-time. Slight delays were observed in certain real-world conditions due to the additional computational load and real-time processing requirements. Overall, the system's responsiveness was adequate, but further optimization may be needed to match the near-instantaneous reactions seen in simulations.

4.3.4.3 Based on Algorithm Efficiency

The simulation results generated high efficiency in obstacle detection and risk assessment algorithms. Moreover, accurate prediction of potential collision scenarios and optimal avoidance strategies. The algorithms implemented in the prototype performed effectively, mirroring the simulation results. However, real-world variables introduced complexities not fully accounted for in simulations, such as sensor noise and unexpected obstacles. Continuous tuning and adaptation of the algorithms were necessary to maintain efficiency and accuracy to maintain the effectiveness of the vehicle collision avoidance system.

4.3.4.4 Based on Reliability

Comparing prototype testing results to simulation outcomes based on reliability provides crucial insights into the consistency and dependability of the simulation models in predicting real-world behavior. Reliability comparisons focus on assessing the degree to which the simulation accurately represents the performance and behavior of the vehicle collision avoidance system under different conditions. Observed from the tests, the simulation result once again has the upper hand on this one. The prototype’s reliability is less than the simulation results as expected. This was mainly due to numerous environmental factors that the simulation did not take in consideration such as the road conditions, sensor interfering elements and the response time of the actuators. Discrepancies between prototype testing and simulation results indicate limitations in the simulation model's ability to accurately predict real-world system performance.

4.3.4.5 Based on Integration with Vehicle Systems

Simulation results were observed to involve seamless integration with virtual vehicle models in the simulation environment as well as effective interaction with the simulated vehicle collision avoidance features. The prototype exhibited successful integration with actual vehicle systems, including braking, and speed adjustments. Some initial challenges in achieving flawless communication between the vehicle collision avoidance system and existing vehicle control units were encountered and subsequently addressed.

Chapter Five
Conclusion and Recommendation

5.1 Conclusion

This project focused on the design and simulation of a vehicle collision avoidance system, aiming to enhance road safety and reduce traffic accidents through technological advancements. The integration of sensors, microcontrollers, and actuators with sophisticated algorithms for obstacle detection and collision prevention has shown promising results.

The implementation of a PID controller to manage speed and braking in MATLAB Simulink simulations demonstrated high accuracy in obstacle detection and effective speed control. Although the system's performance was validated through simulations and showed rapid response times and efficient avoidance maneuvers, real-world tests revealed inconsistencies under complex conditions. This indicates the need for further optimization and real-world validation to ensure the reliability of the system.

5.2 Recommendation

The successful development of the vehicle collision avoidance system project has laid a strong foundation, but there are several avenues for future work and enhancement. For instance, further controller refinement such as continued refining of the PID controller parameters through additional iterations and advanced tuning methods. Sensor calibration is also another consideration that could be used to reduce differences between simulated and actual positions. Another is actuator dynamics enhancement to better capture the real-world behavior of the vehicles.

Moreover, investigate the application of machine learning algorithms and artificial intelligence techniques to enhance the vehicle collision avoidance system capabilities. Develop algorithms for predictive analytics, pattern recognition, and decision-making to improve collision prediction accuracy and response effectiveness. Additionally, integrate features such as automatic emergency steering and warning systems on the vehicle collision avoidance system.

References

[1] Al-Smadi, Adnan M; Wasan, Al-Ksasbeh; Mohammad, Ababneh, "Intelligent automobile collision avoidance and safety system.," *nternational Multi-Conference on Systems, Signals & Devices (SSD),* vol. 19, no. 1, pp. 319-322, 2020.

[2] Rakhonde, Mahesh A., S. A. Khoje, and R. D. Komati., "Vehicle collision detection and avoidance with pollution monitoring system using IoT.," *Global Conference on Wireless Computing and Networking (GCWCN),* no. 2, pp. 75-79, 2018.

[3] P. e. a. Ramya, "Development of a Driver Fatigue and Intoxication Detection System Using Raspberry Pi and Ultrasonic Sensors.," *International Journal of Advanced Engineering Research and Science,* vol. 9, no. 3, pp. 123-135, 2022.

[4] Aditi Padayar, Dipali Jadhav, Priti Pashte, Shweta Lagade, Prof. S. K. Srivastava, "Microcontroller -based Accident Prevention System Using IOT," *Proceedings of the International Conference on Engineering Innovations*, vol. 5, no. 4, pp. 78-85, 2021.

[5] Mubashir Murshed, Md Sanaullah Chowdhury, "IoT-based Car Accident Prevention and Detection System with Smart Brake Control," *International Journal of Advanced Research in Computer Science*, vol. 11, no. 5, pp. 123-135, 2020.

[6] Rachita Shettar, Sandeep Dabhade, Basavaraj Viraktamath, Amit Dalal, Varsha B Vannur, "Design and Development of Accident Prevention and Control Systems," *International Journal of Vehicle Safety and Control*, vol. 15, no. 6, pp. 345-356, 2025.

[7] S. M. e. a. Manaa, "Design and Implementation of a Vehicle Collision Avoidance System Using Modern Sensor Technologies and Microcontrollers.," *International Journal of Advanced Engineering Research and Science,* vol. 8, no. 7, pp. 123-135, 2021.

[8] A. S. Jamaludin, "Development and Implementation of a Collision Avoidance System Using Sensors and Automated Braking Mechanisms," *Proceedings of the International Conference on Robotics and Automation*, vol. 15, no. 8, pp. 78-85, 2020.

[9] M. B K, " Development of a Low-Cost Vehicle Collision Avoidance System Using Camera Technology," *International Journal of Advanced Engineering Research and Science,* vol. 9, no. 9, pp. 234-245, 2022.

Appendix

Atmega 328p Datasheet

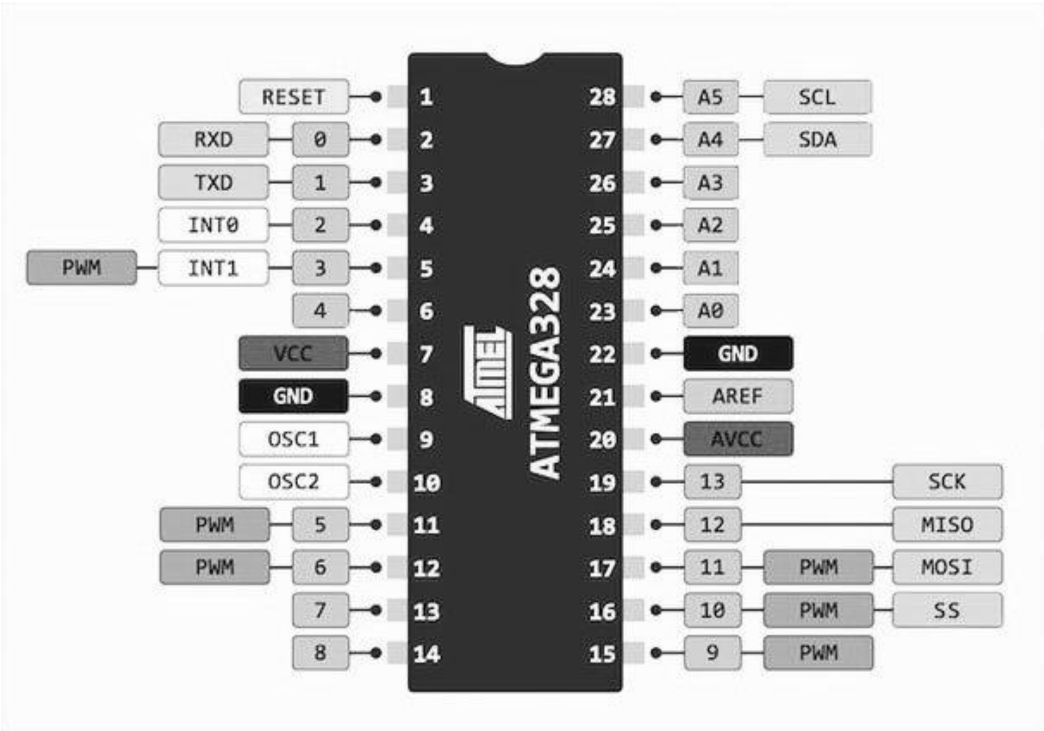


Figure app 1. Atmega328p Microcontroller

The Atmega328p microcontroller pins were selected and configured by following the pinout datasheet below:

Table app 1. ATMega328 Pinout Configuration

Pin No.	**Pin name**	**Description**	**Function**
1.	PC6 (RESET)	Pin6 of PORTC	Pin by default is used as RESET pin. PC6 can only be used as I/O pin when RSTDISBL Fuse is programmed.
2.	PD0 (RXD)	Pin0 of PORTD	RXD (Data Input Pin for USART) USART Serial Communication Interface [Can be used for programming]
3.	PD1 (TXD)	Pin1 of PORTD	TXD (Data Output Pin for USART)

			USART Serial Communication Interface [Can be used for programming] INT2 (External Interrupt 2 Input)
4.	PD2 (INT0)	Pin2 of PORTD	External Interrupt source 0
5.	PD3 (INT1/OC2B)	Pin3 of PORTD	External Interrupt source1 OC2B (PWM - Timer/Counter2 Output Compare Match B Output)
6.	PD4 (XCK/T0)	Pin4 of PORTD	T0 (Timer0 External Counter Input) XCK (USART External Clock I/O)
7.	VCC		Connected to positive voltage
8.	GND		Connected to ground
9.	PB6 (XTAL1/TOSC1)	Pin6 of PORTB	XTAL1 (Chip Clock Oscillator pin 1 or External clock input) TOSC1 (Timer Oscillator pin 1)
10.	PB7 (XTAL2/TOSC2)	Pin7 of PORTB	XTAL2 (Chip Clock Oscillator pin 2) TOSC2 (Timer Oscillator pin 2)
11.	PD5 (T1/OC0B)	Pin5 of PORTD	T1(Timer1 External Counter Input) OC0B (PWM - Timer/Counter0 Output Compare Match B Output)
12.	PD6 (AIN0/OC0A)	Pin6 of PORTD	AIN0(Analog Comparator Positive I/P) OC0A (PWM - Timer/Counter0 Output Compare Match A Output)
13.	PD7 (AIN1)	Pin7 of PORTD	AIN1(Analog Comparator Negative I/P)
14.	PB0 (ICP1/CLKO)	Pin0 of PORTB	ICP1(Timer/Counter1 Input Capture Pin) CLKO (Divided System Clock. The divided system clock can be output on the PB0 pin)
15.	PB1 (OC1A)	Pin1 of PORTB	OC1A (Timer/Counter1 Output Compare Match A Output)
16.	PB2 (SS/OC1B)	Pin2 of PORTB	SS (SPI Slave Select Input). This pin is low when controller acts as slave. [Serial Peripheral Interface (SPI) for programming]

			OC1B (Timer/Counter1 Output Compare Match B Output)
17.	PB3 (MOSI/OC2A)	Pin3 of PORTB	MOSI (Master Output Slave Input). When controller acts as slave, the data is received by this pin. [Serial Peripheral Interface (SPI) for programming] OC2 (Timer/Counter2 Output Compare Match Output)
18.	PB4 (MISO)	Pin4 of PORTB	MISO (Master Input Slave Output). When controller acts as slave, the data is sent to master by this controller through this pin. [Serial Peripheral Interface (SPI) for programming]
19.	PB5 (SCK)	Pin5 of PORTB	SCK (SPI Bus Serial Clock). This is the clock shared between this controller and other system for accurate data transfer. [Serial Peripheral Interface (SPI) for programming]
20.	AVCC		Power for Internal ADC Converter
21.	AREF		Analog Reference Pin for ADC
22.	GND		GROUND
23.	PC0 (ADC0)	Pin0 of PORTC	ADC0 (ADC Input Channel 0)
24.	PC1 (ADC1)	Pin1 of PORTC	ADC1 (ADC Input Channel 1)
25.	PC2 (ADC2)	Pin2 of PORTC	ADC2 (ADC Input Channel 2)
26.	PC3 (ADC3)	Pin3 of PORTC	ADC3 (ADC Input Channel 3)
27.	PC4 (ADC4/SDA)	Pin4 of PORTC	ADC4 (ADC Input Channel 4) SDA (Two-wire Serial Bus Data Input/output Line)

			SDA is used to transmit data from or to a target device.
28.	PC5 (ADC5/SCL)	Pin5 of PORTC	ADC5 (ADC Input Channel 5) SCL (Two-wire Serial Bus Clock Line) SCL is a serial clock primarily controlled by the controller device. SCL is used to synchronously clock data in or out of the target device.

Arduino Codes

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27,16,2);  // set the LCD address to 0x27 for a 16
chars and 2 line display

const int enA = 9;

const int in1 = 5;

const int in2 = 6;

 // Motor B connections

const int enB = 10;

const int in3 = 7;

const int in4 = 8;

#define trigPin 11

#define echoPin 12

  float duration, distance;

void setup(){

  Serial.begin(9600);

  // Motor control pins are outputs

  pinMode(enA, OUTPUT);

  pinMode(enB, OUTPUT);

  pinMode(in1, OUTPUT);

  pinMode(in2, OUTPUT);

  pinMode(in3, OUTPUT);

  pinMode(in4, OUTPUT);

  pinMode(echoPin, INPUT);

  pinMode(trigPin, OUTPUT);

  lcd.init();

  lcd.init();
```

```
 lcd.backlight();
}
void loop(){
  lcd.backlight();
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration*.0343)/2;
  Serial.print("Distance: ");
  Serial.println(distance);
  delay(100);
lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("Distance =");
  lcd.setCursor(11,0);
  lcd.print(distance);
  lcd.setCursor(0,1);
  lcd.print("Speed =");
  if (distance >= 60) {
    lcd.backlight();
    lcd.setCursor(7,1);
  lcd.print(180.);
   lcd.setCursor(12,1);
  lcd.print("km/h");
  analogWrite(enA, 255);
```

```
 analogWrite(enB, 255);
 digitalWrite(in1, HIGH);
 digitalWrite(in2, LOW);
 digitalWrite(in3, HIGH);
 digitalWrite(in4, LOW);
   }
   else if (distance < 60 && distance >= 50) {
     lcd.backlight();
 lcd.setCursor(7,1);
 lcd.print(150.);
 lcd.setCursor(12,1);
 lcd.print("km/h");
analogWrite(enA, 150);
 analogWrite(enB, 150);
 digitalWrite(in1, HIGH);
 digitalWrite(in2, LOW);
 digitalWrite(in3, HIGH);
 digitalWrite(in4, LOW);
   }
    else if (distance < 50 && distance >= 40) {
     lcd.backlight();
lcd.setCursor(7,1);
 lcd.print(.80);
  lcd.setCursor(12,1);
 lcd.print("km/h");
 analogWrite(enA, 130);
 analogWrite(enB, 130);
 digitalWrite(in1, HIGH);
```

```
 digitalWrite(in2, LOW);
 digitalWrite(in3, HIGH);
 digitalWrite(in4, LOW);
    }
    else if (distance < 40 && distance > 30) {
     lcd.backlight();
     lcd.setCursor(7,1);
 lcd.print(.30);
  lcd.setCursor(12,1);
 lcd.print("km/h");
analogWrite(enA, 80);
 analogWrite(enB, 80);
 digitalWrite(in1, HIGH);
 digitalWrite(in2, LOW);
 digitalWrite(in3, HIGH);
 digitalWrite(in4, LOW);
    }
    else if (distance < 30 && distance > 20) {
     lcd.backlight();
     lcd.setCursor(7,1);
 lcd.print(.10);
  lcd.setCursor(12,1);
 lcd.print("km/h");
analogWrite(enA, 50);
 analogWrite(enB, 50);
 digitalWrite(in1, HIGH);
 digitalWrite(in2, LOW);
 digitalWrite(in3, HIGH);
```

```
digitalWrite(in4, LOW);
   }
  else if (distance < 20) {
    lcd.backlight();
     lcd.setCursor(7,1);
lcd.print(0.00);
 lcd.setCursor(12,1);
lcd.print("km/h");
  digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
digitalWrite(in3, LOW);
digitalWrite(in4, LOW);
     }
  delay(10); }
```

MATLAB Script

```
% Motor parameters
R = 0.01;       % Resistance (Ohm)
L = 0.01;       % Inductance (H)
Kb = 0.86;      % Back EMF constant (V/rad/s)
Km = 1;         % Torque constant (Nm/A)
J = 50;         % Moment of inertia (kg.m^2)
b = 1.5;        % Damping coefficient (Nm.s)
% Transfer function
numerator = Km;
denominator = [J*L, (J*R + L*b), (b*R + Kb*Km)];
motor_tf = tf(numerator, denominator);
% PID controller parameters
Kp = 60;
Ki = 80;
Kd = 11.25;
% Create PID controller
pid_controller = pid(Kp, Ki, Kd);
% Open-loop system
open_loop_system = series(pid_controller, motor_tf);
% Closed-loop system
closed_loop_system = feedback(open_loop_system, 1);
% Simulate closed-loop step response
t = 0:0.01:5;  % Time vector
[y, t] = step(closed_loop_system, t);
% Open-loop transfer function of the motor
numerator = Km;
denominator = [J*L, (J*R + L*b), (b*R + Kb*Km)];
```

```
motor_tf = tf(numerator, denominator);
% Plot pole-zero map
figure;
pzmap(motor_tf);
title('Pole-Zero Map of the Open-Loop System');
grid on;
set(gca,'color','k')
% Open-loop system
open_loop_system = series(pid_controller, motor_tf);
% Closed-loop system
closed_loop_system = feedback(open_loop_system, 1);
% Plot pole-zero map of the closed-loop system
figure;
pzmap(closed_loop_system);
title('Pole-Zero Map of the Closed-Loop System');
grid on;
set(gca,'color','k')
figure;
rlocus(open_loop_system);
title('Root Locus of the Open-Loop System');
grid on;
set(gca,'color','k')
```