

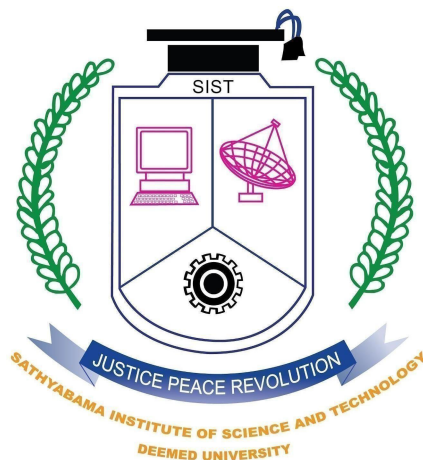
HANGING ROBOT - HIGH VOLTAGE TRANSMISSION LINE INSPECTION ROBOT

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering in Mechatronics

By

THARUN.A (39090050)

SHEGUN.S (39090057)



**DEPARTMENT OF MECHATRONICS
SCHOOL OF MECHANICAL ENGINEERING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade "A" by NACC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600119**

MAY- 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NACC | 12B Status by UGC | Approved by AICTE

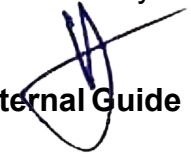
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600119

www.sathyabama.ac.in

DEPARTMENT OF MECHATRONICS

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **THARUN. A (39090057)**, **SHEGUN. S (39090050)** who carried out the project entitled "**HANGING ROBOT - HIGH VOLTAGE TRANSMISSION LINE INSPECTION ROBOT**" under our supervision from Jun to May 2023.


Internal Guide

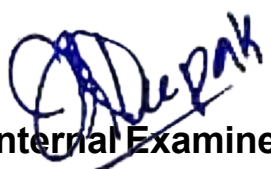
Mr.V. JAYAPRAKASH, M.Tech., MIE., (Ph.D.)



Head of the Department

Dr. S. PRAKASH, M.E., Ph.D.,

Submitted for Viva Voice Examination held on 11th April 2023


Internal Examiner


External Examiner

DECLARATION

We **THARUN. A (39090057)** and **SHEGUN. S (39090050)** hereby declare that the Project Report entitled “**HANGING ROBOT - High Voltage Transmission Line Inspection Robot**” done by us under the guidance of **Mr. V. JAYAPRAKASH M.Tech., MIE., (Ph.D.)** at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree Mechatronics.

1)

S. Shegun S.

2)

Tharun A

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. S. PRAKASH, M.E., Ph.D.**, Dean, School of Mechanical engineering and Head of the Department, Department of Mechatronics for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Mr. V. JAYAPRAKASH, M.Tech., MIE., (Ph.D.)**, for his valuable guidance, suggestions and constant encouragement paved the way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Mechatronics who are helpful in many ways for the completion of the project.

ABSTRACT

By using an Arduino board, as the brains for the device, along with motors an automated vehicle can be designed. However, by complementing it with even one input sensor such as an CURRENT Sensor, the vehicle's functionality increases significantly. In the prototype, we have added various sensor modules like the CURRENT Sensor, Bluetooth module HC-05 and Pressure sensors to allow the prototype to function in different modes. Instead of designing a new vehicle for each independent mode, we have decided to compact all into one device.

There are several different methods of communicating with the prototype which would allow us to change the modes such as Pressure, Wi-Fi and Bluetooth. To reduce the physical interaction with the vehicle and make it as automated as possible, we would be changing the modes by giving instructions to the vehicle via Bluetooth. Thus being able to switch to whatever mode needed effortlessly and conveniently.

In order to be able to give commands to the prototype, we developed a simple Android application to control the speed of the motors as well as change modes with ease.

TABLE OF CONTENT

CHAPTER	TITLE	PAGE NO
	ABSTRACT	I
	LIST OF TABLES	IV
	LIST OF FIGURES	V
	LIST OF SYMBOLS AND ABBREVIATIONS	VI
1	INTRODUCTION	1
2	LITERATURE SURVEY	3
	AIM AND SCOPE OF THE PRESENT INVESTIGATION	6
	3.1 INTERFERENCE FROM LITERATURE	6
	3.2 PROPOSED OBJECTIVES	6
	3.3 ORGANISATION OF THE THESIS	7
4	EXPERIMENTAL SETUP AND METHODOLOGY	8
	4.1 METHODOLOGY	8
	4.2 OPERATION	9
	4.3 EXPERIMENTAL SETUPS	10
	4.1.1 CURRENT SENSOR	11
	4.1.2 NODEMCU	11
	4.1.3 RELAY	12
	4.1.4 DC MOTOR	12
	4.1.5 POWER SUPPLY UNIT	13
	4.1.6 WIFI MODULE	13
	4.4 INTRODUCTION	14
	4.5 HARDWARE	14
	4.6 ARDUINO UNO R3	14
	4.7 OVERVIEW	14
	4.8 CURRENT SENSOR OVERVIEW	16
	4.9 UART	16
	4.10 THE ASYNCHRONOUS RECEIVING AND TRANSMITTING PROTOCOL	17
	4.11 THE PHYSICAL LAYER STANDARDS	18
	4.12 TTL LEVEL UART	18
	4.13 SG-90 SERVO MOTOR	21
	4.14 OVERVIEW	22
	4.15 TYPE OF MOTORS	22
	4.16 BLUETOOTH MODULE HC-05	23
	4.17 OVERVIEW OF IR SENSOR	23
	4.18 MOTOR DRIVER L-298	24
	4.19 OVERVIEW	24
	4.20 SOFTWARE	25
	4.21 INTRODUCTION	25
	4.22 MAIN CODE	25
	4.23 FRONT END	29
	4.24 BACK END	30
	4.25 WORKING	30
	4.26 EXPERIMENTAL SETUP	32
	4.27 CONTROLLING APPLICATION OF	33

	THE PRODUCT	
	4.27.1 BLUETOOTH RC CONTROLLER	33
	4.27.2 BLUETOOTH RC CONTROLLER APPLICATION INTERFACE	33
	4.28 DATA ANALYSIS OF THE CURRENT	33
	4.28.1 THING VIEW FREE	33
	4.28.2 THING VIEW FREE APPLICATION INTERFACE	33
	4.29 WIRE DEFECT DETECTION USING CAMERA BY LIVE TRACKING	34
	4.29.1 IP WEBCAM	34
	4.29.2 IP WEBCAM APPLICATION INTERFACE	34
	CAMERA IN THE PRODUCT	
	4.29.3 IP WEBCAM	35
	4.29.4 IP WEBCAM APPLICATION INTERFACE OF THE INSPECTOR	35
5	RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS	36
	5.1 HARDWARE DESIGN	36
	5.2 RESULTS	38
6	SUMMARY AND CONCLUSIONS	39
	REFERENCES	41

LIST OF TABLES

TABLE NO.	TABLE NAME	PAGE NO
	CHARACTERISTICS OF ATMEGA328 PIN DETAILS	15

LIST OF FIGURES

FIGURE NO	TITLE NAME	PAGE NO
4.1	BLOCK DIAGRAM	8
4.2	CIRCUIT DIGRAM OF THE PROPOSED PROTOTYPE	10
4.3	EXPERIMENTAL SETUP OF PROPOSED PROTOTYPE	11
4.4	CURRENT SENSOR	11
4.5	NODE MCU	12
4.6	RELAY	12
4.7	DC MOTOR	12
4.8	POWER SUPPLY UNIT	13
4.9	WIFI MODULE	13
4.10	ARDUINO UNO R3	14
4.11	ATMEGA328P AND ARDUINO UNO PIN MAPPING	15
4.12	DETECTION OF AN OBJECT BY AN CURRENT SENSOR	16
4.13	UNIVERSAL ASYNCHRONOUS RECEIVER / TRANSMITTER	17
4.14	TTL LEVEL UART	19
4.15	PINOOTS OF THE DB-9	20
4.16	SG-90 SERVO MOTOR	21
4.17	HC-05 MODULE	23
4.18	IR SENSOR	23
4.19	MOTOR DRIVERL298	24
4.20	USER INTERFACE OF APPLICATION	29
4.21	APPLICATION BACK END	30
4.22	EXPERIMENTAL SETUP	33
4.23	BLUETOOTH RC CONTROLLER APPLICATION INTERFACE	33
4.24	THING VIEW FREE APPLICATION INTERFACE	33
4.25	IP WEBCAM APPLICATION INTERFACE	34
4.26	IP WEBCAM APPLICATION INTERFACE	34
4.27	IP WEBCAM APPLICATION INTERFACE	35
4.28	IP WEBCAM APPLICATION INTERFACE	35
5.1	INITIAL BUILD	37
5.2	CUSTOM FABRICATION OF BASE BOARD	38
6.1	FINISHED PROTOTYPE	40

LIST OF ABBREVIATION

SI.NO	ABBREVIATION	MEANING
1	SONAR	SOUND NAVIGATION AND RANGING
2	V	SPEED
3	T	TEMPERATURE (IN °C)
4	LCD	LIQUID CRYSTAL DISPLAY
5	RISC	REDUCED INSTRUCTION SET COMPUTER
6	ISP	IN-SYSTEM PROGRAMMING
7	EEPROM	ELECTRICALLY ERASABLE PROGRAMMABLE READ-ONLY MEMORY
8	SRAM	STATIC RANDOM-ACCESS MEMORY
9	SPI	SERIAL PERIPHERAL INTERFACE
10	A/D	ANALOG-TO-DIGITAL
11	USB	UNIVERSAL SERIAL BUS
12	MIPS	MICROPROCESSOR WITHOUT INTER-LOCKED PIPELINED STAGES
13	M/S	METRE PER SECOND
14	HZ	HERTZ

CHAPTER 1

INTRODUCTION

An autonomous vehicle can be implemented for various use cases. Depending on the task at hand and the functionality required a set path, action, or group of actions can be defined and coded to deploy an autonomous vehicle. As the fundamental idea is to automate manual processes, the environment the vehicle will be deployed in plays a crucial role in defining the action logic.

Power line inspection and maintenance are fields of application where robotics has yet to be introduced. However, as in many other hostile environments, substantial benefits in terms of workers safety and quality of inspection results could arise. The transmission line inspection robot presents one of the latest initiatives toward that goal. A smart GSM based fault detection and location system was used to adequately and accurately indicate and locate the exact spot where fault had occurred. This will ensure a shorter response time for the technical crew to rectify these faults and thus help save transformers from damage and disasters. The system implements automatic fault detection by including a clamp-meter for current measurement, ultrasonic sensors for sag and obstacle detection, Wi-Fi module and a current sensor for crack detection. The heart of the robot prototype is the ESP8266 microcontroller that coordinates all the sensors to the final working of the robot. It has both to and fro movement that helps in continuous monitoring of the transmission line. Hence, ultrasonic sensors are provided for both forward and backward obstacle detection along with the sag detection sensor provided beneath the body of the microcontroller. A clamp-meter is provided for the accurate current measurement. A current sensor is provided for crack detection. Wi-Fi module helps in providing the user with accurate fault location at the time of fault detection. Finally, the fault information is transmitted to the control room. In conclusion, the time required to locate a fault is drastically reduced, as the system automatically provides accurate fault location information. The main goal of designing such a vehicle is to use it in real world applications such as transportation services, delivery systems, logistics, unmanned operations to name a few. The vehicles can perform desired tasks without continuous human guidance. With increasing demand in automating processes and tasks across various sectors, this simple application-based-

implementation can help with basic understanding of building a stand-alone vehicle to withstand varied terrain and track and deliver the expected functionality while taking into consideration the design aspect of the vehicle as well.

CHAPTER 2

LITERATURE SURVEY

“OBSTACLE AVOIDING HANGING ROBOTIC VEHICLE USING ARDUINO”;

This paper describes the design and implementation of an obstacle- avoiding hanging robotic vehicle using Arduino.

“LOW COST ARDUINO WIFI BLUETOOTH INTEGRATED PATH FOLLOWING HANGING ROBOTIC VEHICLE WITH WIRELESS GUI REMOTE CONTROL”;

This paper presents a low-cost Arduino-based hanging robotic vehicle that can follow a path and be controlled remotely using WiFi and Bluetooth.

“PRESSURE LINE FOLLOWING AND CURRENT SENSOR NAVIGATING HANGING ROBOT WITH ATMEGA328 PRO”;

This paper describes a hanging robot that uses pressure sensors and current sensors for line following and navigation, respectively.

“OBJECT FOLLOWER AND BARRIER ESCAPING HANGING ROBOT USING IMAGE PROCESSING”; This paper discusses a hanging robot that can follow objects and avoid barriers using image processing techniques.

“BLUETOOTH CONTROLLED HANGING ROBOT”; This paper presents a hanging robot that can be controlled using Bluetooth.

A low-cost obstacle avoidance hanging robot, This paper describes a hanging robot that uses pressure sensors and current sensors for line following and navigation, respectively. low-cost hanging robot that can avoid obstacles.

"provides a datasheet for the ATMEGA328P microcontroller used in the design of such robots." This paper describes the design and implementation of an obstacle-avoiding hanging robotic vehicle using Arduino.

"describes the design and development of an interface module for obstacle-avoiding hanging robots," This paper describes the design and development of an interface module for obstacle-avoiding hanging robots.

"discusses proximity measurement techniques used in mechatronics." This paper provides a datasheet for the ATMEGA328P microcontroller used in the design of hanging robots..

"both pertain to the Arduino software and its integration into hanging robotic vehicle designs," This paper describes the design and implementation of an obstacle-avoiding hanging robotic vehicle using Arduino.

"discusses obstacle detection and avoidance for an autonomous surface vehicle." This paper describes a hanging robot that uses pressure sensors and current sensors for line following and navigation, respectively.

"describes the design and implementation of an obstacle-avoiding hanging robotic vehicle using Arduino," This paper discusses proximity measurement techniques used in mechatronics.

"presents a pressure line following and current sensor navigating hanging robot."This paper describes a hanging robot that uses pressure sensors and current sensors for line following and navigation, respectively.

"discusses a Bluetooth-controlled hanging robot,"This paper presents a hanging robot that can be controlled using Bluetooth.

"describes an object follower and barrier-escaping hanging robot using image processing techniques."This paper describes the design and implementation of an obstacle-avoiding hanging robotic vehicle using Arduino.

CHAPTER 3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

Its ultimate purpose is to inspect the transmission lines. through cameras and send the data to the ground. The robot consists of two motors, four wheels, sensors, cameras etc. This robot can cross the clamps on the lines. The transmission lines provide additional power to pump stations, which help move our product through the pipelines. Transmission lines are large, high voltage lines that typically deliver power to smaller distribution lines that go to homes, businesses and other buildings. The primary reason that power is transmitted at high voltages is to increase efficiency. As electricity is transmitted over long distances, there are inherent energy losses along the way. High voltage transmission minimizes the amount of power lost as electricity flows from one location to the next. The real purpose of transmission line protection is to detect faults abnormal operating conditions and to initiate corrective action. Protective relays must be able to evaluate a lot of parameters to choose and establish the right corrective action. Obviously, a relay cannot prevent the fault.

3.1 INTERFERENCE FROM LITERATURE

The different papers used as reference for this project have been selected for the algorithms used to implement the logic for control of the vehicle. Integrating different functionality into one switchable mode poses as a challenge of the algorithm used to switch and control different modes if they are varied. Picking up the basic implementation logic from various papers helped us in combining and delivering our prototype with the desired functionality.

3.2 PROPOSED OBJECTIVES

The primary goal of this undertaking is to design an automated vehicle with multiple modes such as:

- Path Following Mode
- Obstacle Avoiding Mode
- Voice Control Mode
- Bluetooth Remote Control Mode

In addition, command the vehicle prototype to switch between modes as well as vary speed of the motors of the prototype wirelessly via Bluetooth. This will be

achieved with the help of an application that sets up communication between the prototype and the Android phone using Bluetooth as a means. Thereby controlling the vehicle remotely as well as switching to the desired mode of function.

3.3 ORGANISATION OF THE THESIS

This thesis comprises of five chapters. First chapter discusses about the objectives, theory and literature reviews of the project. Chapter two will talk about the technique that has been done to finish this venture. It will explain details about the devices used to complete the circuit and the software.

Construction of the circuit and its working will be presented in chapter three. Chapter four will consist of the results of the prototype. Last but not least, chapter five will discuss the conclusion and recommendation that can be done for the future work

CHAPTER 4

EXPERIMENTAL SETUP AND METHODOLOGY

4.1 METHODOLOGY

The block diagram mainly consists of

- Current Sensor
- NodeMCU
- Relay
- DC Motor
- Power supply unit
- Wifi Module

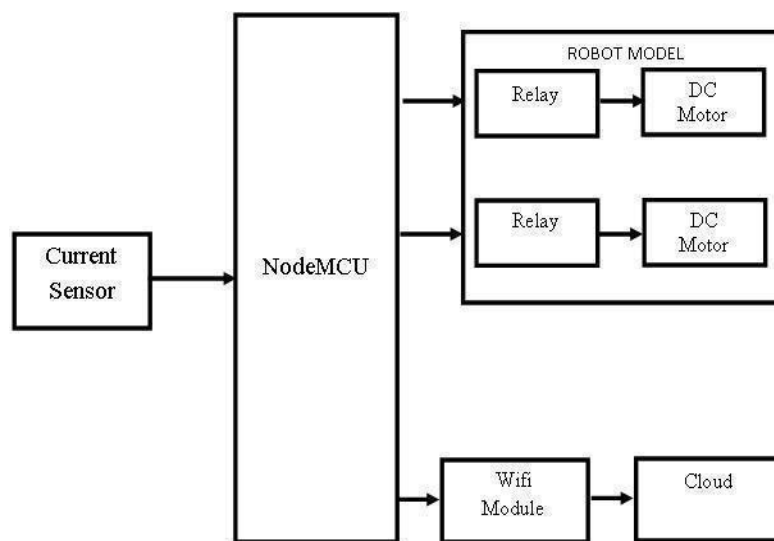


Fig:4.1:Block Diagram

According to the block diagram, 230V supply is given to the ESP8266 microcontroller, through an adapter. It is the heart of the system and is chosen so as to aid the robot movements. All the sensors are interfaced with the microcontroller which coordinates all the functions. An Current sensor is used for the same and it detects the cracks on the pathway and sends the information to the microcontroller for further action. The meter helps to measure the current and display the value of current at the meter display. The current hikes initially and finally rests at a value of 1A. The forward and backward movement of the motor is facilitated by the gear mechanism alongwith the dcmotor. The robot prototype is

programmed to follow forward path for at least 1m and then returns its path after the limit.

4.2 OPERATION

The circuit diagram implements the NodeMCU for the controlling action. All the sensors are connected to the corresponding Vcc and ground. The Txd and Rxd pins of Wi-Fi module are connected to the ESP8266 via, port pins. The sensors sense the various faults like sag, obstacle etc and send the information to the ESP8266 microcontroller. The robotic mechanism is also controlled by the same controller. It is facilitated by the gear mechanism with the aid of a dc motor. The components used in the making of this prototype have been specifically considered with their respective ratings and functionality in mind.

The sensors, motors, mounting brackets, Arduino, Motor driver shield etc. have been packaged and mounted on the chassis to effectively use the space at offer, making the vehicle compact and easy to handle.

All the connections seen above are the basic signal and power pins used for each component connected to the Arduino. Li-ion Battery pack (3000mAh) cells are used to supply the peak load from all the sensors and components working simultaneously. The maximum rated rpm at the highest configured speed for each motor is 255rpm and can be varied to 0rpm is necessary. This alteration in speed can be achieved directly through a slider control within the Application. Direct access buttons have been configured to switch between the different modes of operation and directly communication to the Bluetooth module and update the Code loop. The circuit is connected to ensure no reconfiguration or unexpected load is shared to any sensor and damage the circuit continuity.

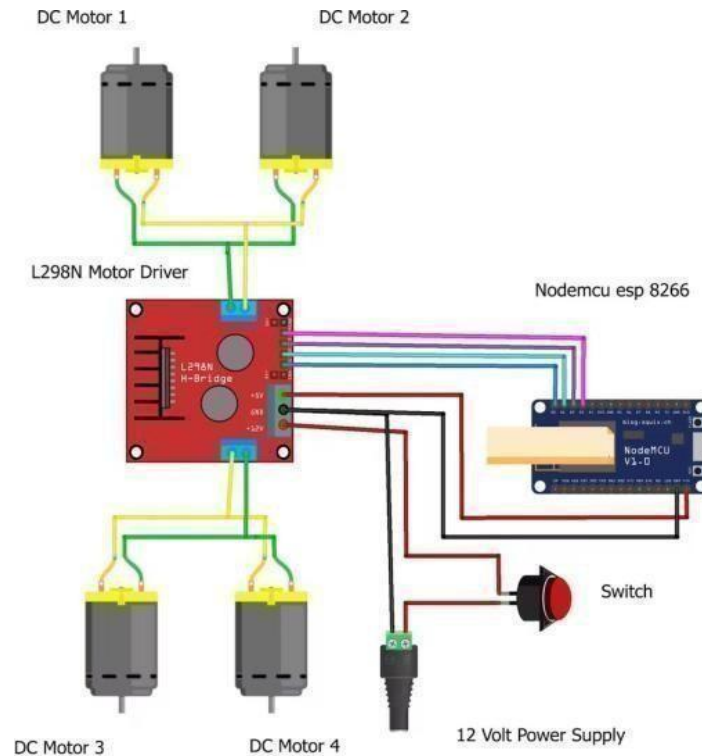


Fig:4.2: Circuit diagram of the proposed prototype.

A single phase induction motor is used as the load to the line. The power supply circuit to the ESP8266 is also shown in the figure and it helps to stepdown the voltage to 3.3V which is acceptable to the ESP8266 .

4.3 EXPERIMENTAL SETUPS

The figure shows the experimental setup of the Transmission line inspection robot. It consists of ESP8266 microcontroller, Wi-Fi module, Current sensor, clamp meter, DC motor, single phase induction motor. As the robot prototype runs along the line, it can detect the obstacles, current fluctuations, abnormal sag and convey the information about the fault to the control room, I.e. user mobile phone. This is achieved with the help of a Wi-Fi module that sends the message regarding the fault location and the type of fault. After a predetermined distance (i.e. 1m), the robot prototype stops and reverses it's motion to continuous inspection along the same line. The setup includes two wooden poles that support a straight 1m Al rod of 6mm diameter which acts as the sample transmission. The robot moves along the rod using gear mechanism which is actuated by a dc motor. The rod is supplied by a single phase induction motor, which is clamped on both sides of the rod as shown in figure. The current through

the rod is measured and displayed on a clamp on meter attached to the body of the robot. Current hikes through a series of value and finally settles at a reading of one ampere. This steady state current and their by the load can be varied by using a motor of higher rating. As the prototype moves along the line, the current fluctuations from the rated 1A is detected by a clamp on meter and displayed on its screen. Any obstacle along the line will be detected by the ultrasonic sensor and the robot prototype stops, reverses its motion away from the obstacle and corresponding message send to control room.

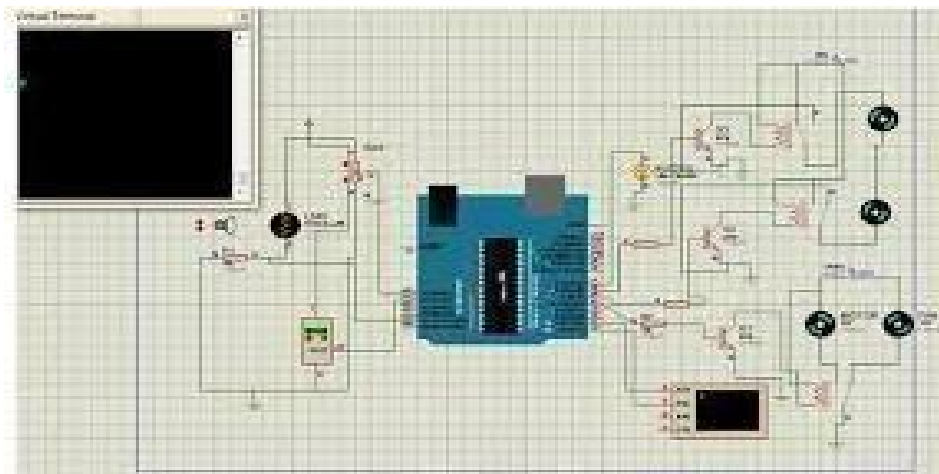


Fig:4.3: Experimental Setup of Proposed prototype

4.1.1 Current Sensor

Current sensors, also commonly referred to as current transformers or CTs, are devices that measure the current running through a wire by using the magnetic field to detect the current and generate a proportional output. They are used with both AC and DC current.



Fig:4.4:Current Sensor

4.1.2 NodeMCU

The name "NodeMCU" combines "node" and "MCU" (microcontroller unit). Strictly speaking, the term "NodeMCU" refers to the firmware rather than the associated development kits. Both the firmware and prototyping board designs are

open source. The firmware uses the Lua scripting language.



Fig:4.5:NodeMCU

4.1.3 Relay

A relay is the device that opens or closes the contacts to cause the operation of the other electric control. Relays control one electrical circuit by opening and closing contacts in another circuit



Fig:4.6:Relay

4.1.4 DC motor

A DC motor is an electrical machine that converts electrical energy into mechanical energy. In a DC motor, the input electrical energy is the direct current which is transformed into the mechanical rotation.

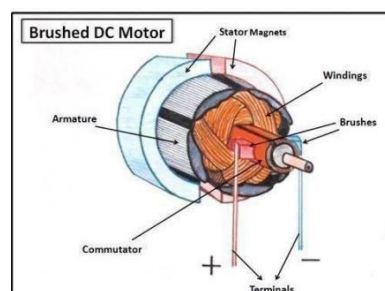


Fig:4.7:Dc Motor

4.1.5 Power supply unit

A power supply unit (PSU) is a hardware device that converts AC electricity into DC electricity and then distributes it to the rest of the computer. On a standard desktop computer, the PSU is where the power cord plugs into and usually has an I/O power switch on it

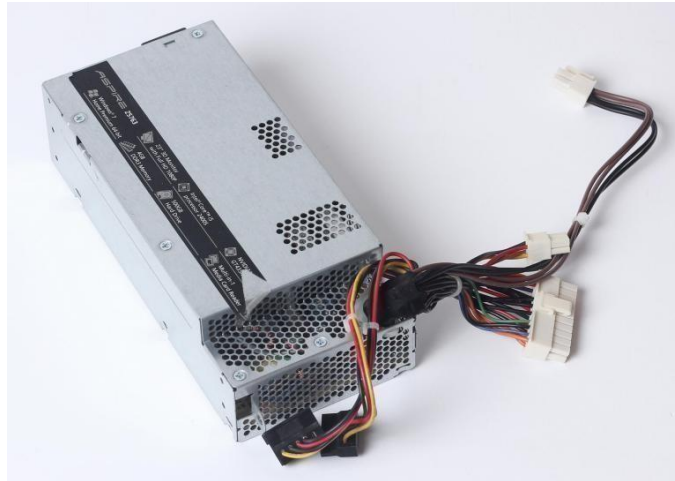


Fig:4.8:Power Supply Unit

4.1.6 Wifi Module

WiFi modules (wireless fidelity) also known as WLAN modules(wireless local area network) are electronic components used in many products to achieve a wireless connection to the internet.



Fig4.9:Wifi Module

4.4 INTRODUCTION

This section gives clarification on the technique and step utilized in structuring the prototype vehicle. It incorporates the depiction of the design flow, structure engineering and detail clarifications for the product advancement as well as equipment improvement of the prototype

4.5 HARDWARE

4.6 Arduino Uno R3



Figure 4.10: Arduino Uno R3

4.7 Overview

A versatile microcontroller which is widely used throughout for various electronic projects is the Arduino Uno dependent on the ATmega 328P. It has computerized i/o pins , 6 analogous inputs, a Universal Serial Bus, and a reset button. Comprises of all that anticipated to assist the microcontroller; interface Personal Computer as well as provide power to via a battery or a connector. There are various applications for which an Arduino board can be used. Such as:

- Modelling prototypes and systems of electronic nature
- Endless possibilities of DIY (Do-It-Yourself) Undertakings
- Simple beginner level usage
- Undertakings requiring multiple I/O pins

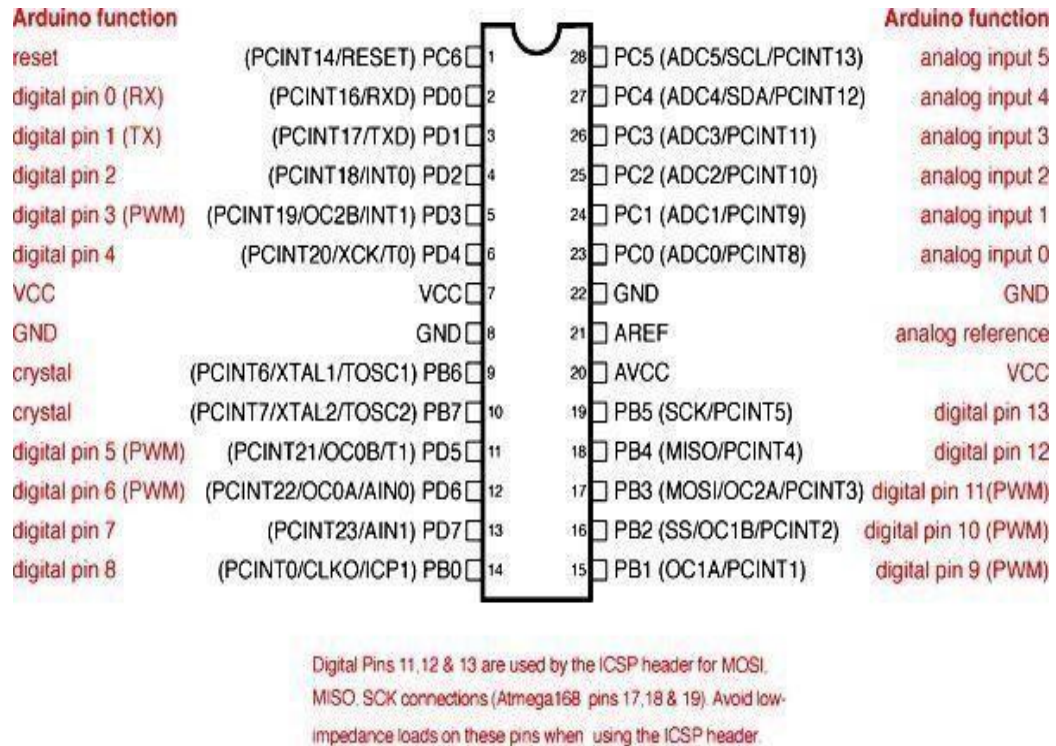


Figure 4.11: ATmega328P and Arduino Uno Pin Mapping

The Arduino Uno is AVR RISC centered board, which operates sandwiched between 5.5 Volts, conglomerates as much as nearly 32 kb memory, three versatile timers/counters with compare modes, wire serial interface, programmable watchdog timer, 1 KB EEPROM, 2 KB SRAM, and power saving modes. To program the Arduino Uno board, the IDE is to be used. Arduino can be used to communicate with a computer, another Arduino board or supplementary micro controllers. There are two LEDs on the Arduino board which will flash to indicate when data is being transmitted. A SoftwareSerial library allows for transfer of data on any of its digital pins.

Table 4.1: Characteristics of AT Mega 328P

Parameter	Value
CPU type	8-bit AVR

Performance	20 MIPS @ 20 MHz
SRAM	2 KB
Pin count	28 / 32 pin
Max. operating frequency	20 MHz
No. of touch channels	16
Max. I/O pins	23

4.8 CURRENT SENSOR OVERVIEW

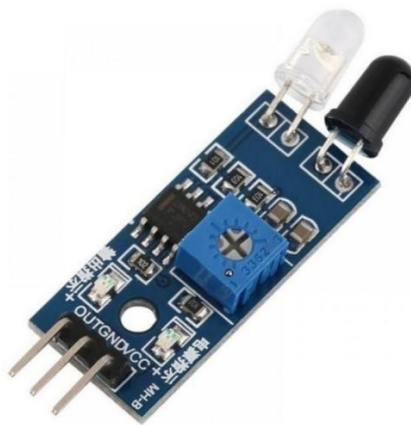


Fig4.12:Detection of an object by an CURRENT Sensor

4.9 UART

The Universal Asynchronous Receiver/Transmitter (UART) controller is the key component of the serial communications subsystem of a computer. UART is also a common integrated feature in most microcontrollers. The UART takes bytes of data and transmits the individual bits in a sequential fashion. At the destination, a second UART re-assembles the bits into complete bytes. Serial transmission of digital information (bits) through a single wire or other medium is

much more cost effective than parallel transmission through multiple wires. Communication can be “full duplex” (both send and receive at the same time) or “half duplex” (devices take turns transmitting and receiving).

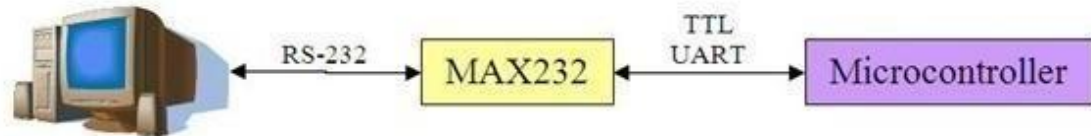


Fig4.13: Universal Asynchronous Receiver/Transmitter

A UART (Universal Asynchronous Receiver/Transmitter) is the microchip with programming that controls a computer's interface to its attached serial devices. Specifically, it provides the computer with the RS-232C Data Terminal Equipment (DTE) interface so that it can "talk" to and exchange data with modems and other serial devices. As part of this interface, the UART also:

- Converts the bytes it receives from the computer along parallel circuits into a single serial bit stream for outbound transmission
- On inbound transmission, converts the serial bit stream into the bytes that computer handles
- Adds a parity bit (if it's been selected) on outbound transmissions and checks the parity of incoming bytes (if selected) and discards the parity bit
- Adds start and stop delineators on outbound and strips them from inbound transmissions
- Handles interrupts from the keyboard and mouse (which are serial devices with special ports)
- May handle other kinds of interrupt and device management that require coordinating the computer's speed of operation with device speeds.

4.10 THE ASYNCHRONOUS RECEIVING AND TRANSMITTING PROTOCOL

Asynchronous transmission allows data to be transmitted without the sender having to send a clock signal to the receiver. In this case, the sender and receiver must agree on timing parameters (Baud Rate) prior transmission and special bits are added to each word to synchronize the sending and receiving units. In asynchronous transmission, the sender sends a Start bit, 5 to 8 data bits (LSB first), an optional Parity bit, and then 1, 1.5 or 2 Stop bits. When a word is passed

to the UART for asynchronous transmissions, the Start bit is added at beginning of the word. The Start bit is used to inform the receiver that a word of data is about to be send, thereby forcing the clock in the receiver to be in sync with the clock in the transmitter. It is important to note that the frequency drift between these two clocks must not exceed 10%. In other words, both the transmitter and receiver must have identical baud rate. After the Start bit, the individual bits of the word of data are sent, beginning with the Least Significant Bit (LSB). When data is fully transmitted, an optional parity bit is sent to the transmitter. This bit is usually used by receiver to perform simple error checking. Lastly, Stop bit will be sent to indicate the end of transmission.

When the receiver has received all of the bits in the data word, it may check for the Parity Bits (both sender and receiver must agree on whether a Parity Bit is to be used), If the Stop Bit does not appear when it is supposed to, the UART considers the entire word to be garbled and will report a Framing Error to the host processor when the data word is read.

4.11 THE PHYSICAL LAYER STANDARDS

There are actually quite a number of different standards that utilizes similar protocol. For instances, TTL level UART, RS-232, RS-422, RS-485 and etc. We will only discuss about TTL level UART and RS-232 in this article.

4.12 TTL level UART

Most microcontrollers with UART use TTL (Transistor-transistor Logic) level UART. It is the simplest form of UART. Both logic 1 and 0 are represented by 5V and 0V respectively.

The TTL level UART is commonly used in the communications between microcontrollers and ICs. Only 2 wires are required for the full duplex communications as illustrated in the pictu below.

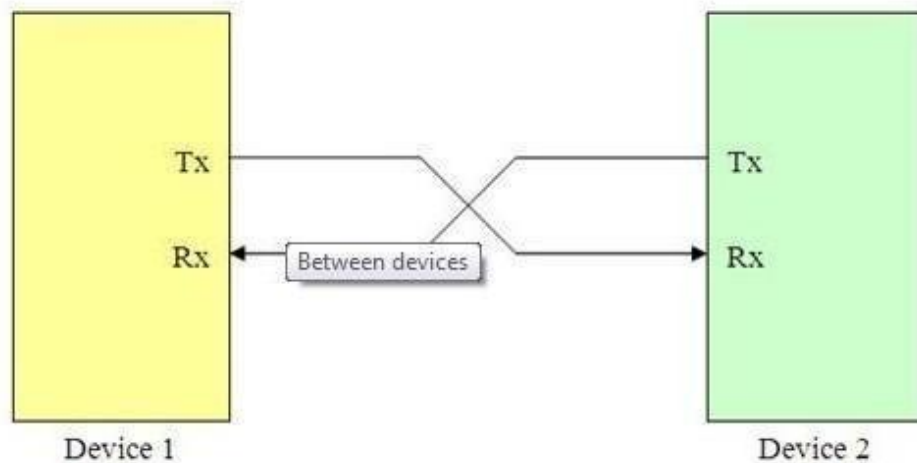


Fig4.14:TTL level UART

RS-232 (Recommended Standard 232) is a standard for serial binary data signals connecting between a Data Terminal Equipment (DTE) and a Data Communication Equipment (DCE). It is commonly used in computer serial ports. One of the significant differences between TTL level UART and RS-232 is the voltage level. Valid signals in RS-232 are ± 3 to ± 15 V, and signals near 0V is not a valid RS-232 level.

Logic	Voltage
Low	+3 to +15V
High	-3 to -15V

voltage level for RS-232

Besides voltage level, the RS-232 also has a few extra pins specifically designed for the communication between PC and modem. The pinouts of the DB-9 and their functions are shown below.

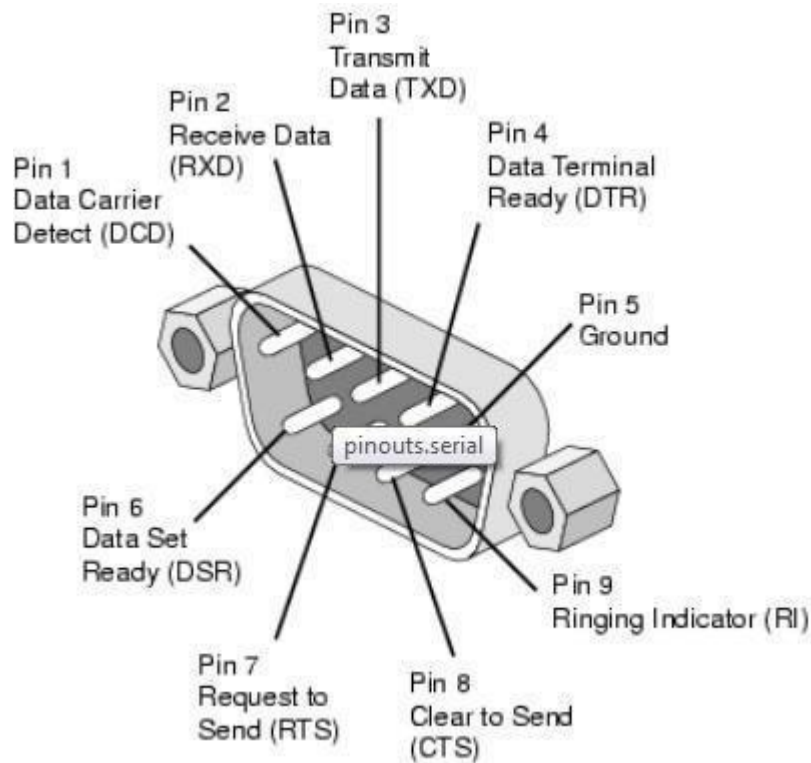


Fig.4.15:pinots of the DB-9

PINOUTS AND DESCRIPTION OF A SERIAL PORT

Name	Pin	Description
TransmittedData (TxD)	3	Serial data output
Received Data(RxD)	2	Serial data input
Request to Send (RTS)	7	This line informs the DCE (Modem) thatthe DTE (PC) is ready to exchange data
Clear to Send(CTS)	8	This line indicates that the DCE is ready to exchange data
Data TerminalReady (DTR)	4	Asserted by DTE to indicate that it isready to be connected
Data Set Ready	6	Asserted by DCE to indicate the DCE is

(DSR)		powered on and is ready to receive commands or data for transmission from the DTE
Data Carrier Detect (DCD)	1	Asserted by DCE when a connection has been established with remote equipment
Ring Indicator (RI)	9	Asserted by DCE when it detects a ring signal from the telephone line

4.13 SG-90 Servo Motor



Fig4.16: SG-90 Servo Motor

4.14 Overview

It is a turning electrical gadget that transforms electrical energy (Direct Current) into mechanical energy. A loop of coil inside the motor delivers a field of magnetisation that makes revolving movement when a voltage (DC in nature) provided to its ends. There is also a shaft made of iron present within the motor and is engulfed by a loop of wire. The aforementioned shaft consists of two immovable magnets (North and South) on the two sides which results in both a repulsive and attracting force, thereby delivering torque.

4.15 Type of motors

The following are examples of direct current motors:

- Brushed Motor
- Brushless Motor
- Gear Motors (Planetary)
- Gear Motors (Spur)
- Stepper Motors
- Coreless Brushless Motors
- Servo Motor
- Gear head Motor

Gear Motors – common electrical motors transform electrical energy into mechanical energy .

A gear motor is an amalgamation of a motor and gearbox. The addition of a gear head to a motor reduces the speed and increases the torque output. The most important parameters in regards to gear motors are:

- Speed (rpm rating)
- Torque (lb/in)
- Efficiency (%)

In order to select the appropriate gear motor for our application we must analyse:

- Load
- Speed
- Torque

4.16 Bluetooth Module HC-05

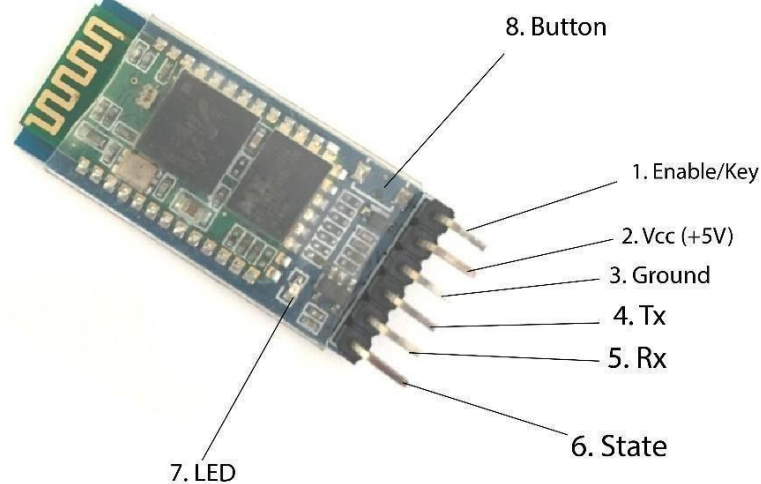


Figure 4.17: HC-05 Module

4.17 Overview OF IR SENSOR

The HC-05 is a module which can give a remote functionality which is widely used to establish communication between devices or speak with any gadget via Bluetooth functionality example a laptop or mobile. There exist a variety of android apps. that are now accessible which makes this cycle significantly simpler. The components speak with the comfort of USART at 9600 baud rate henceforth not difficult to interface with any device that is compatible with USART.

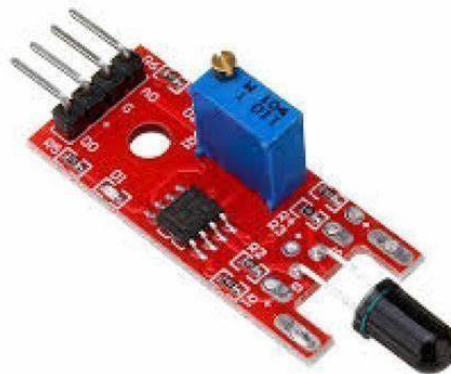


Fig4.18:IR Sensor

4.18 Motor Driver L-298



Figure 4.19: Motor Driver L298

4.19 Overview

L298N Motor Driver Shield is a high capacity module for driving DC and StepperMotors. This module consists of L298 motor driver IC and a 78M05 5V regulator.L298N Module can govern up to 4 DC motor, or 2 DC motors with direction andspeedcontrol.

Applications of L298N Module:

Drive DC motors. Drive stepping motors

In Hanging robotics

4.20 SOFTWARE

4.21 Introduction

Arduino projects are written in the Arduino Integrated Development Environment (IDE). Arduino IDE is an exceptional program running on a system that empowers the user to create outlines for different Arduino sheets.The Arduino programming language relies upon a direct hardware programming language called processing, which resembles the C language. The sketch is then executed by the board after it is uploaded to it from the IDE.

4.22 Main Code

```
char junk;
String inputString="";void conditions(); void sendmsg();void gps();
#include <SoftwareSerial.h> SoftwareSerial mySerial(10, 11); // RX, TXvoid setup()
{
  Serial.begin(9600); pinMode(5,OUTPUT);pinMode(6,OUTPUT);
  pinMode(7,OUTPUT);pinMode(8,OUTPUT);mySerial.begin(9600);
}
void loop()
{
  if(mySerial.available()){ while(mySerial.available())
  {
    char inChar = (char)mySerial.read();inputString += inChar;
  }
  mySerial.println(inputString); while (mySerial.available() > 0)
  {
    junk = mySerial.read() ;}
  if(inputString == "B")
  {

    digitalWrite(4, HIGH);digitalWrite(5, LOW):
    digitalWrite(6, HIGH);digitalWrite(7, LOW);
  }
  else if(inputString == "F")
  {
    digitalWrite(4, LOW);digitalWrite(5, HIGH);digitalWrite(6, LOW);digitalWrite(7,
    HIGH);
  }
  else if(inputString == "R")
  {
    digitalWrite(4, HIGH);digitalWrite(5, LOW);digitalWrite(6, LOW);digitalWrite(7,
    HIGH);
  }
}
```

```

else if(inputString == "L")
{
digitalWrite(4, LOW);digitalWrite(5, HIGH);digitalWrite(6, HIGH);digitalWrite(7,
LOW);
}
else if(inputString == "V")
{
digitalWrite(4, LOW);digitalWrite(5, LOW);digitalWrite(6, LOW);digitalWrite(7,LOW);
}
else if(inputString == "U")
{
digitalWrite(4, LOW);digitalWrite(5, LOW); digitalWrite(6, LOW);digitalWrite(7,
LOW);
int pir=analogRead(A0);if(pir>500)
{
sendmsg();
}
}
else
{
digitalWrite(4, LOW);digitalWrite(5, LOW);digitalWrite(6, LOW);digitalWrite(7,LOW);
}
inputString = "";
}

void sendmsg()
{
Serial.println("AT"); delay(2000); Serial.println("AT+CMGF=1");delay(2000);
Serial.println("AT+CMGS=\"+919360722154\\r\"");delay(1000);
Serial.println("Human Detected at");delay(1000);
gps(); delay(2000); Serial.println((char)26);delay(4000);
Serial.println("AT"); delay(2000); Serial.println("AT+CMGF=1");delay(2000);

```

```

Serial.println("Human Detected at");delay(1000);
Serial.println("AT+CMGS=\"+919360722154\\r\"");delay(1000);
gps(); delay(2000); Serial.println((char)26);delay(4000);
}
void gps()
{
/*String byteGPS=Serial.read();if (byteGPS == -1)
{
delay(100);
} else {

linea[conta]=byteGPS;conta++;if (byteGPS==13)
{
cont=0;bien=0; for (int i=1;i<7;i++)
{
if (linea[i]==comandoGPR[i-1])
{
bien++;

}

}

if(bien==6)
{
for (int i=0;i<300;i++){if (linea[i]=='')
{
indices[cont]=i;cont++;
}
if (linea[i]=='*')
{
indices[12]=i;cont++;
}
}
}
}
}

```

```

Serial.println("");
Serial.println(""); for (int i=2;i<5;i++)
{
if(i==2)
{
for (int j=indices[i];j<(indices[i+1]-1);j++)
{
((linea[j+1]));
}}
if(i==4)
{
Serial.print(",");
for (int j=indices[i];j<(indices[i+1]-1);j++)
{
((linea[j+1]));
}}
sendmsg();
}
}
conta=0;
for (int i=0;i<300;i++){linea[i]=' ';
}
}
}*/
}

```



Figure 4.20: User Interface of Application

4.23 FRONT END

As the buttons Connect and Disconnect pertain to establishing communication between the phone and the Bluetooth module attached to the prototype. In between these two buttons, the status of the connection is presented. If the communication is established and a success, the status bar will read “Connected” else will read “Disconnected”.

The arrow keys are present to control the vehicle manually by tapping said buttons. In the centre is the icon with a microphone on it. This is the Voice control mode which utilises the Google speech recognition software and interprets specific commands such as “Go Forward”, “Go Backward”, “Turn Left”, and “Turn Right” which can be given into the microphone of the phone by anyone.

On the bottom of the application, there are three buttons which are as follows:

1. LF – Switch to Line Following mode
2. Manual – Switch to Remote Control (RC) mode
3. OA – Switch to Obstacle Avoiding mod

Apart from these buttons, there is a slider below the mode changing buttons. This slider allows the user to change the speed at which the prototype will

function in RC mode. This varies from 0 RPM to 255 RPM.

4.24 Back End

Using the MIT App developer site, we constructed the back end of the application. As the site is easy to use and accessible, the backend is in block structure.

As can be seen in the below figure 2.12, the backend of the application though immense, is in the form of blocks that can be connected to one another simply by dragging and clicking. This is the feature provided by the MIT App Inventor which allows anyone to create Android applications.

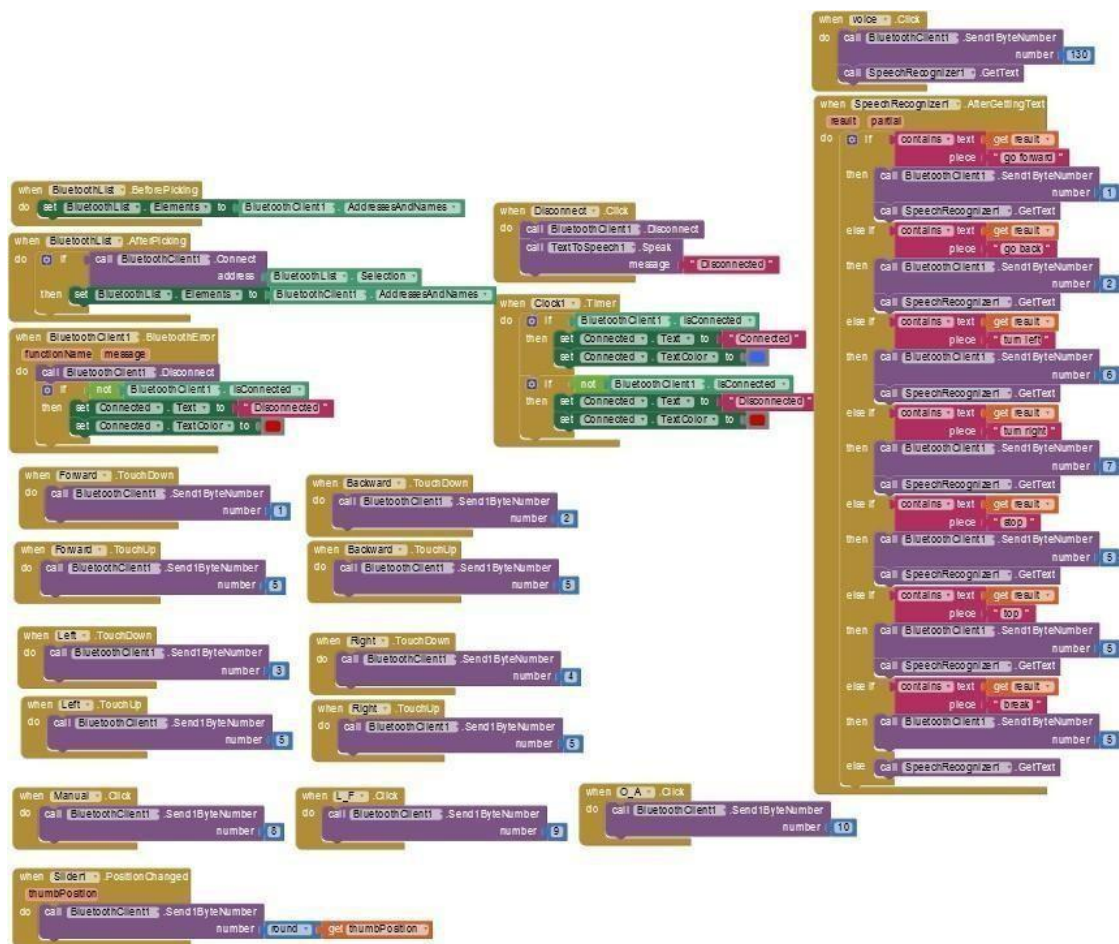


Figure 4.21: Application Back end

4.25 WORKING

Bluetooth is the key to this entire integration and prototype. Once the power is in and all the sensors are available, the vehicle can be connected to the application by first connecting the Bluetooth module to the device of control i.e. Smartphone and then navigating within the app for the final step. Once completed the interface shows the connected notification right in the middle. Once

connected the vehicle can operate in 4 distinct modes. Voice Controlled Mode
Bluetooth Control (RC) Mode Line-Following Mode Obstacle Avoiding Mode

Each mode can be switched between using the dedicated button control to toggle between the same. Once a particular mode is under use, the vehicle will solely follow the instruction set of that particular logic unless forced to switch into manual mode. Manual mode is the default mode for the vehicle.

Input from the application is fed to the module which instructs the Arduino to send out appropriate signals to control the movement of all four motors individually. The Arduino board is the main controller for all the functions. Every input signal from the Bluetooth module is sent to the Arduino and decoded based on the specific mode it is configured to work in that instance. The signals are processed on FIFO logic. Any change in the Application or while execution of a specific command will complete the loop and go into default manual mode to avoid and unexpected movement or uncontrolled manoeuvres.

Motor connection are controlled per side of the vehicle. Left control is independent from the Right control. Front and Back motors are controlled through one connection point and operate in sequence to control from and back wheel motion of each side individually. In case of overlapping signal to the motors, the default configuration is to perform the series of action based on FIFO logic. If the input signal is not sent correctly the motor does not accept the signal and Motor Driver Shield will produce a distress signal due to excess load on the battery or overlapping commands. In case of insufficient power to the motors, the sensors will make a distress noise which can indicate and help troubleshoot the issue quickly. All the modes cannot be configured to work simultaneously.

4.26 EXPERIMENTAL SETUP

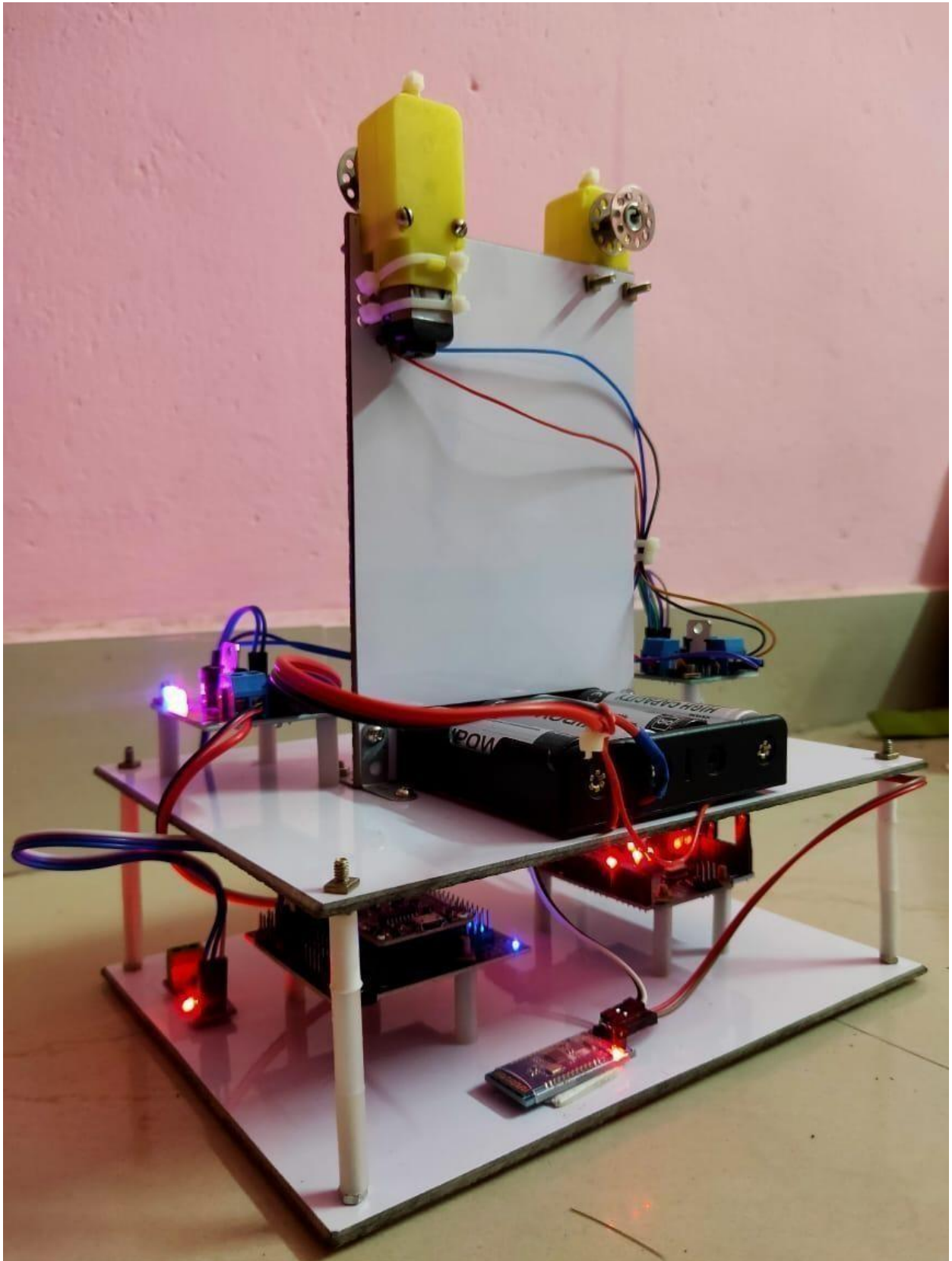


Fig4.22:Experimental setup

4.27 CONTROLLING APPLICATION OF THE PRODUCT

4.27.1 Bluetooth Rc Controller

The application allows you to control a micro controller and Bluetoothfitted RC car with your smart phone

4.27.2 Bluetooth Rc Controller Application Interface

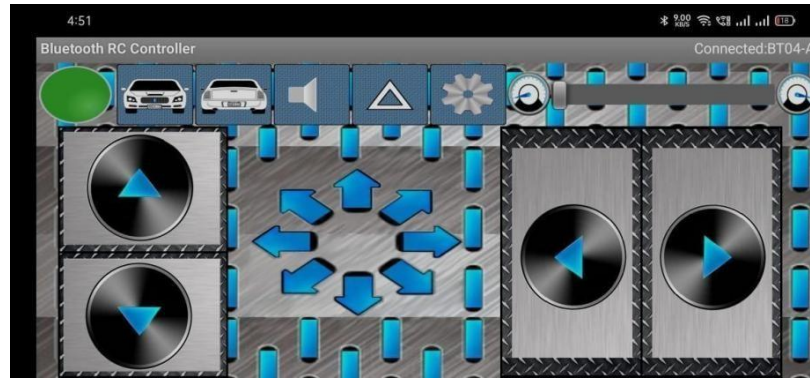


Fig4.23:Bluetooth Rc Controller Application Interface

With this application we are controlling our project high voltage transmission lineinspection robot.and its connected with the bluetooth module which is connected to the bluetooth of our smart phone

4.28 DATA ANALYSINS OF THE CURRENT

4.28.1 Thing View Free

ThingSpeak is an IoT analytics platform service that allows you to aggregate, visualize, and analyze live data streams in the cloud. You can send data to ThingSpeak from your devices, create instant visualization of live data, and send alerts.

4.28.2 Thing View Free Application Interface

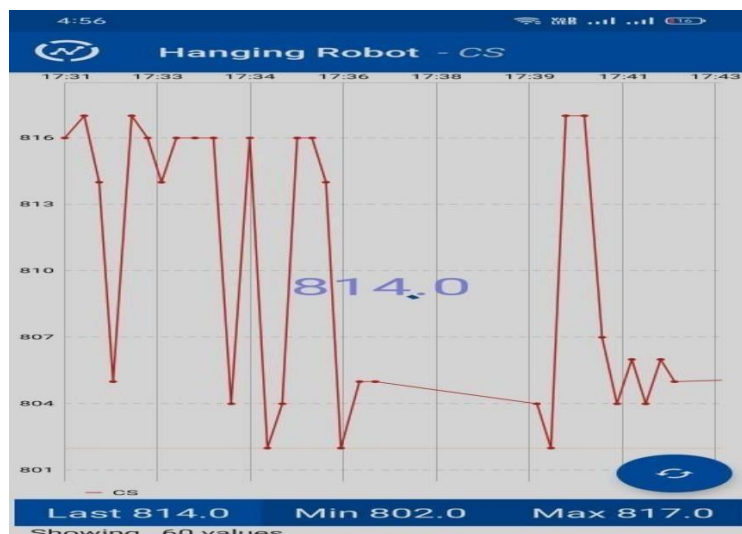


Fig4.24:Thing View Free Application Interface

These are the data collected from the current sensor in the form of graph and numbers.

4.29 WIRE DEFECT DETECTION USING CAMERA BY LIVE TRACKING

4.29.1 Ip Webcam

IP Webcam turns your phone into a network camera with multiple viewing options. View your camera on any platform with VLC player or web browser. Stream video inside WiFi network without internet access.

Optional Ivideon cloud broadcasting is supported for instant global access. Two-way audio supported in tinyCam Monitor on another android device. Use IP Webcam with third-party MJPG software, including video surveillance software, security monitors and most audio players.

4.29.2 Ip Webcam Application Interface camera in the product



Fig4.25:Ip Webcam Application Interface

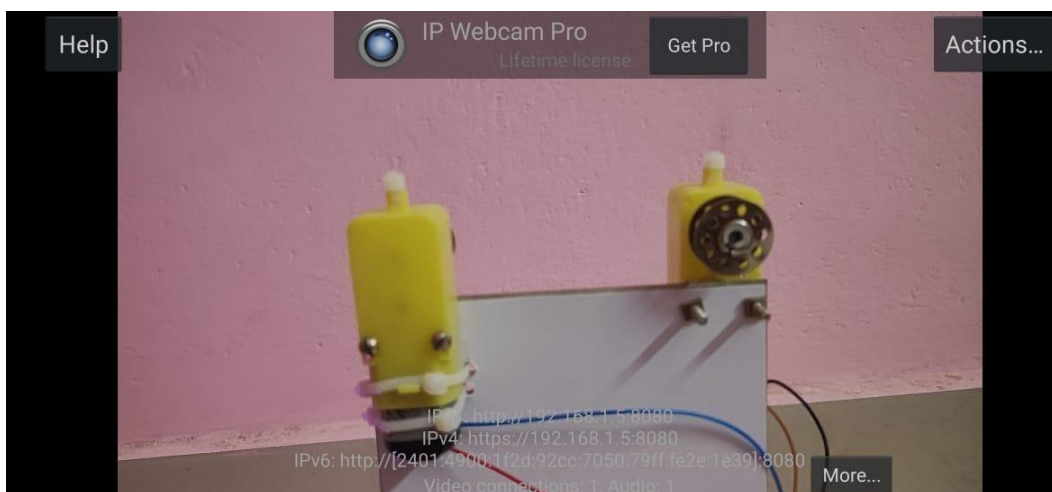


Fig4.26:Ip Webcam Application Interface

4.29.3 Ip Webcam

IP Webcam turns your phone into a network camera with multiple viewing options. View your camera on any platform with VLC player or web browser. Stream video inside WiFi network without internet access. Optional Video cloud broadcasting is supported for instant global access. Two-way audio supported in tiny Cam Monitor on another android device. Use IP Webcam with third-party MJPG software, including videosurveillance software, security monitors and most audio players.

4.29.4 Ip Webcam Application Interface Of The Inspector

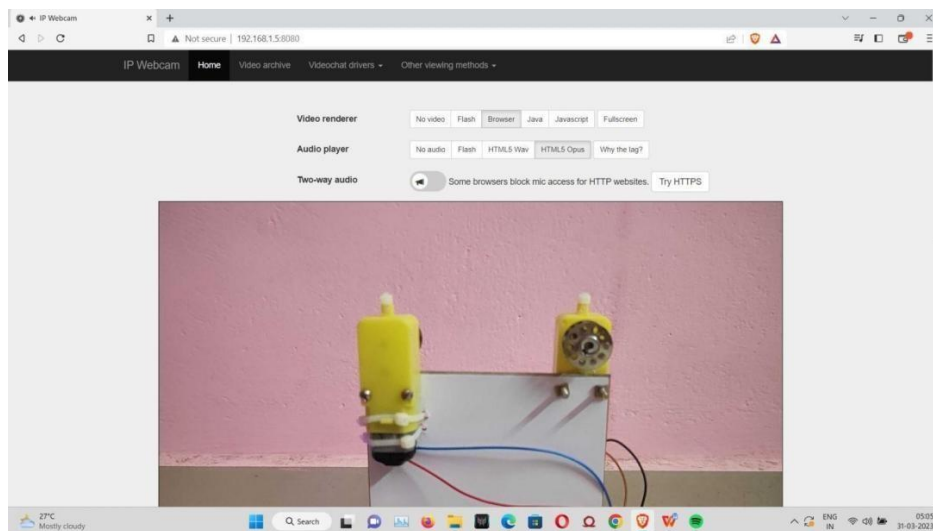


Fig4.27:Ip Webcam Application Interface

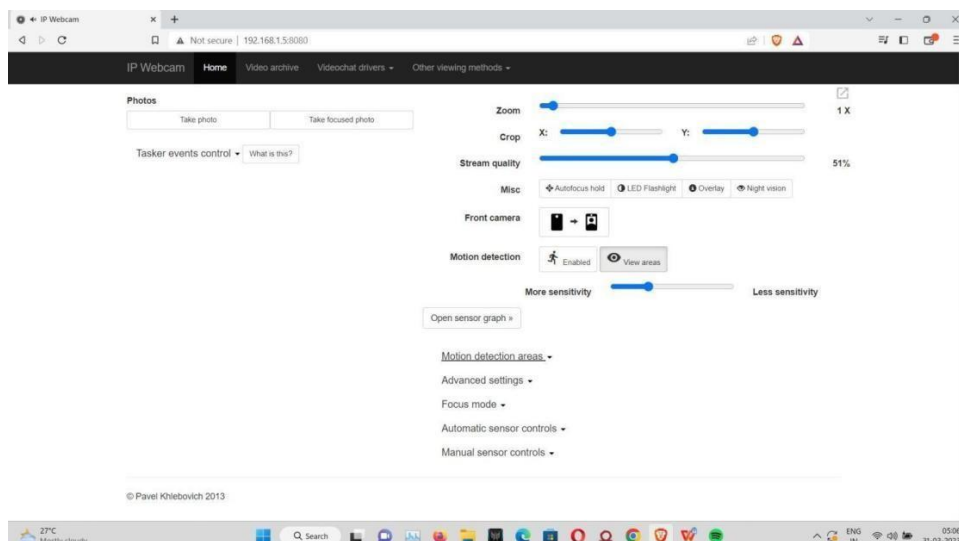


Fig4.28:Ip Webcam Application Interface

CHAPTER 5

RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS

5.1 HARDWARE DESIGN

The hardware design built has been the primary focus to ensure the robustness and durability of the prototype. Base chassis is a Plastic Composite selected to mount all the electronic components without any electrical interference from a board. Precision cut holes were drilled out in the chassis to pass along the wiring between the components placed on the bottom and top of the board. Securing the connection were also a part of the design evaluation.

Extensive use of strong adhesive and mounts to ensure all the components do not move around and cause any damage to other components around it. Packaging the vehicle in this small chassis involved multiple iteration and changes to ensure the level of durability we aimed for. Motor Driver, Arduino Uno and all the individual components, i.e. the CURRENT Sensor, Bluetooth HC-05 Module and the Pressure Sensor, have been either connected or soldered with the help of a soldering iron.

Every connection made is to ensure the continuity of the circuit is maintained for all the operating modes in one sequence hence eliminating the need for multiple wires or connection for individual modes. Each sensor terminal is distinguished with distinct positive, negative, and Signal connection to make troubleshooting easy and quick. In case of any distress signal from the Arduino or the Motor Driver Shield, the connection terminals can be quickly checked to eliminate the issue. Mounting of motors are measured setup to provide tight manoeuvring control and agile movement of the vehicle in varied environments. Ground clearance and width of the vehicle has been considered as an essential part of the design to make the package as compact as possible. For varied condition and requirements the dimension, load carrying capacity, speed and other such factors can be modified and adjusted for the necessary application



Figure 5.1: Initial Build

5.2 RESULTS

The desired mode of functionality for the vehicle are as follows:

- Bluetooth RC mode (Manual)
- Path Following mod
- Obstacle avoiding mode
- Voice Control Mode

All the modes have delivered the desired functionality in varied environments and terrain as tested. The application developed to interact with the module and control its functionality, built with the intention to keep the UI simple and intuitive. Direct and easy connectivity to the module with low signal communication via Bluetooth ensures no signal is missed or interfered.

Individual Mode functionality as tested delivers the expected results which made the integration of the code logic simple and direct. Any interruption in the connection or change in mode during operation will stop complete the current execution command and switch to manual mode.

The Hardware design goal and requirement have been completed and meet the expected standard to test the prototype build in various environments. Controlled NVH and impact resistance of the structure have also been tested against various surfaces and objects to ensure the functioning of the prototype is not compromised in challenging conditions.



Figure 5.2: Custom Fabrication of Base board

CHAPTER 6

SUMMARY AND CONCLUSIONS

The prototype built to inculcate various different functionalities such as Obstacle Avoidance, Line Following, Bluetooth control, Voice Control showcases various environments such vehicle can be deployed in. Automating manual process by using the base of this vehicle principle can open a wide variety of use cases this implementation can be deployed in.

Minimising physical interaction with the vehicle and automating the process of switching between modes serves as a boon for manual processes that can now be controlled remotely. The performance of the prototype built also highlights the importance of durability and robustness to help improve the longevity and functionality of such vehicle in different scenarios.

The concept of this prototype can be implemented in with various logics to better suit the controllability as per the defined use case, operational hours, frequency of interaction, degree of automation etc. The model showcases the added benefits of inculcating modes of operation as an intuitive, easy to control, and wireless option to increase the functionality of such vehicles which can be deployed in controlled environments to help in improving and reducing manual efforts for various applications.

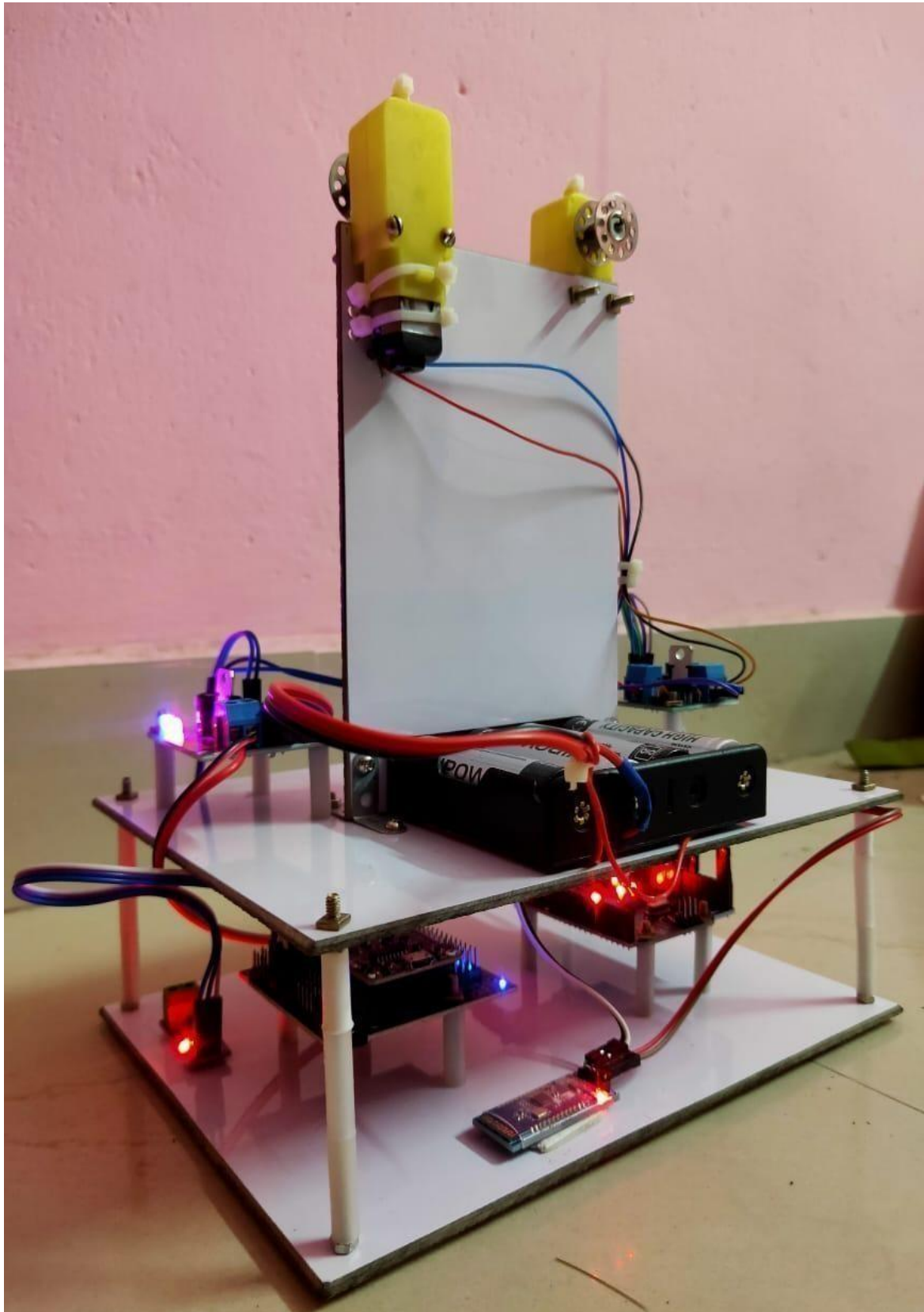


Figure 6.1: Finished Prototype Vehicle

REFERENCES

- [1] V. Hanumante et al., "Low Cost Obstacle Avoidance Hanging robot" in International Journal of Soft Computing and Engineering, Vol. 3, Issue-4, 2013
- ATMEGA328P Datasheet, ATMEL Corporation, San Jose, CA, 2008
- [2] M. Nasucha, "Design and Development of an Interface Module for Obstacle Avoiding Hanging robots", Microcontroller Lab, Univ. Pembangunan Jaya, Tangerang Selatan, Indonesia, 2015
- [3] M. Jouaneh, "Proximity Measurement" in Fundamentals of Mechatronics, SI Edition, United States: Nelson Engineering, ch.7, sec. 7.4, pp. 221-229, 2012
- [4] Arduino. (2015). Arduino Software (IDE). (Arduino) Retrieved Decembe27, 2015, from <https://www.arduino.cc/en/Guide/Environment>
- [5] Heidarrsson, H. K., & Sukhatme, G. S. (2011). "Obstacle Detection and Avoidance for an Autonomous Surface Vehicle using a Profiling Sonar". 2011 IEEE International Conference on Hanging robotics and Automation Shanghai.
- [6] Akрати Pithode, Harsha Wagadre, Neha Verma, Riya Rathore, Vaishnavi Bhangе, Amey Joshi "OBSTACLE AVOIDING HANGING ROBOTIC VEHICLE USING ARDUINO", 2020 International Journal of Engineering Applied Sciences and Technology
- [7] Supantha Mandal, Suraj Kumar Saw, Shilpi Maji, Vivek Das, Sravanth kumar Ramakuri
- [8] Sanjay kumar, "LOW COST ARDUINO WIFI BLUETOOTH INTEGRATED PATH FOLLOWING HANGING ROBOTIC VEHICLE WITH WIRELESS GUI REMOTE CONTROL", 2016, International Conference On Information Communication And Embedded System
- [9] Yuhang Tian, Gengwu Du, "Pressure Line Following and CURRENT SENSOR Navigating Hanging robot with ATMEGA328Pro", 2019, IEEE 3rd Advanced Information Management, Communicates, Electronic and Automation Control Conference.
- [10] Sagar Pramanik, Harendra Kerketta, Dibas Ghosh, Jivesh Kumar Jha. "BLUETOOTH CONTROLLED HANGING ROBOT",

2016, International Journal of Scientific & Engineering Research.

[11] Thennavarajan Subramanian , Rahul Madbhavi , Shreepad Potadar ,Sylvester JeromeD'souza , K V Gangadharan, "OBJECT FOLLOWER AND BARRIER ESCAPINGHANGING ROBOT USING IMAGE PROCESSING", 2015, International Journal of InnovativeResearch in Science, Engineering and Technology.