CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The purpose of robotics in commercial & residential intention has come to be quite essential for executing challenging work into more conveniently simple way. There are a lot of researches working on to enhance the connection between humans and robot. This project describes the implementation of a Rocker bogie mechanism based voice-controlled rover with turn indicator using Arduino. In this project, the user gives specific voice commands to the robot through an Android app installed on the smartphone. At the receiving side, a Bluetooth transceiver module receives the commands and forwards them to the Arduino on the robotic car. Arduino controls the movements of the robot according to received commands. The robot moves forwards, backwards, left and right, and stops according to the voice commands forward, backward, left, right and stop, respectively .Voice recognition system uses selected spoken words as the input, this input allows the machine to identify specific words and subsequently process the particular command and execute it. It is basically a interface between man and robot it receives the voice as input signal and processes it and then forwarded to central controller then it passes the signal to Robot. This method helps the users to perform the task more efficiently and removes the part of humans to manually operate the system. Several applications currently use voice control to perform various tasks. For example, voice-controlled robotic arms are used in the medical field to perform small surgeries. Voice controlled robots have also been explored in other areas such as intelligent wheelchairs, educational learning systems, healthcare systems, and personal communications etc.

1.2 ROCKER BOGIE MECHANISM

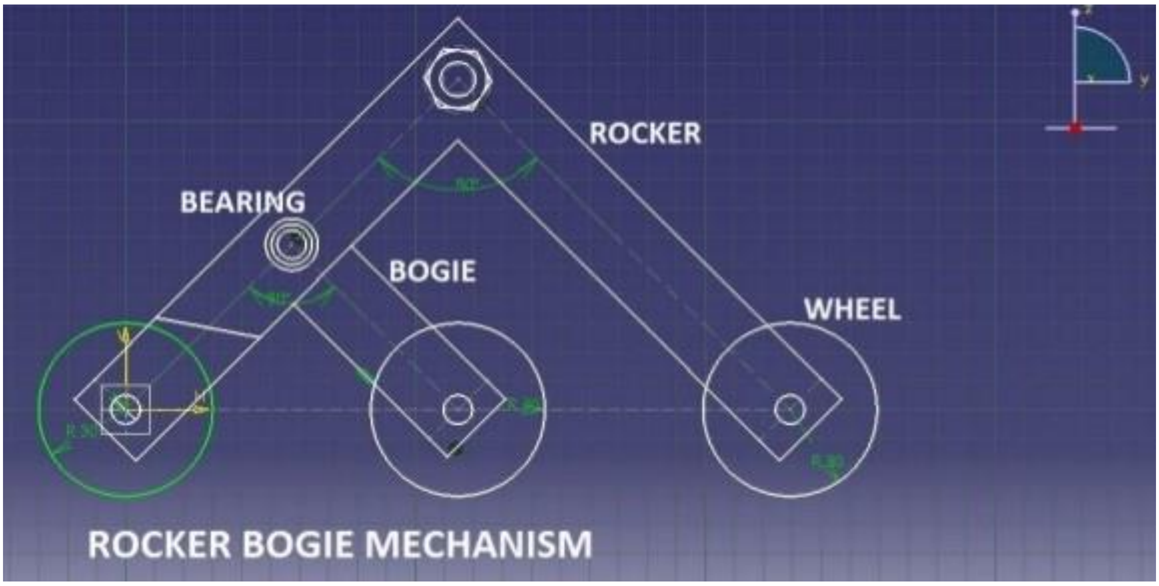


Fig 1.1 Rocker bogie mechanism

The rover uses a rocker-bogie suspension system to drive smoothly over uneven terrain. It has a total of six wheels. Each of six wheels has its own motor. There is one rocker-bogie assembly on each side of rover. The rocker is the larger link. It connects to the rover body (the chassis) in the middle(at the rocker pivot) ,has a wheel in the front and connects to the bogie in the back.The bogie is the smaller link that connects to the rocker in the middle(at the bogie pivot)and has wheels at both ends. The two rockers connect to the body through a mechanism called differential. The differential keeps the body level of the rover. Relative to the body ,when one rocker goes up ,the other rocker goes down. The rocker-bogie design has no springs or stub axles for each wheel, allowing the rover to climb over obstacles (such as rocks) that are up to twice the wheel's diameter in size while keeping all six wheels on the ground. In order to go over a vertical obstacle face, the front wheels are forced against the obstacle by the center and rear wheels. The rotation of the front wheel then lifts the front of the vehicle up and over the obstacle. The middle wheel is then pressed against the obstacle by the rear wheels and pulled against the obstacle by the front until it is lifted up and over. Finally, the rear wheel is pulled over the obstacle by the front two wheels. During each wheel's traversal of the obstacle, forward progress of the vehicle is slowed or completely halted. The system is designed to be used at slow speed of around 10 centimeters per second (3.9 in/s) so as to minimize dynamic shocks and consequential damage to the vehicle when surmounting sizable obstacles. The maximum speed of the robots operated in this way is limited to eliminate as many dynamic effects as possible so that the motors can be geared down, thus enabling each wheel to individually lift a large portion of the entire vehicle's mass. It has been used in the 2003 Mars Exploration Rover mission robots Spirit and Opportunity on the 2012 Mars Science Laboratory (MSL) mission's rover Curiosity, and is slated for use in the Mars 2020 rover.

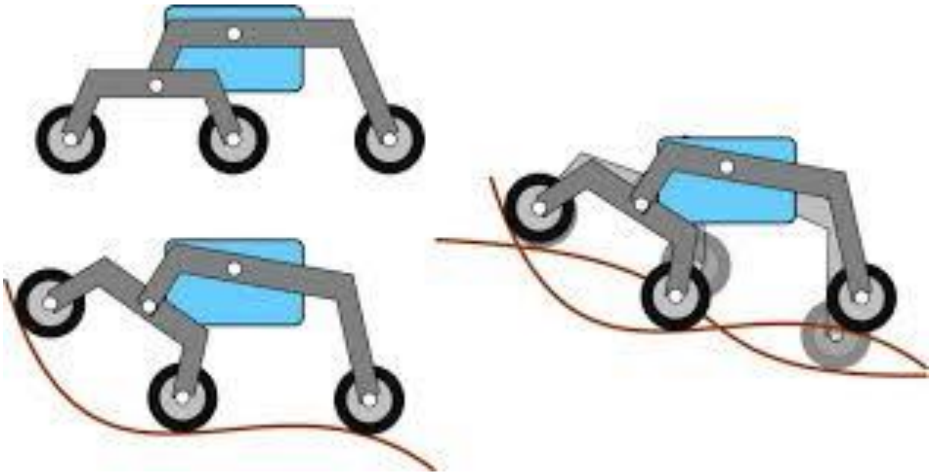


Fig 1.2 Movement of Rocker mechanism over uneven terrain

CHAPTER 2

LITERATURE SURVEY

The concept of our project is to create a Automated Rover, which has a Rocker-Bogie Mechanism that works on speech recognition and is supplemented by turn indicators. The following papers have been studied for analysing and implementing the project. Voice Recognition/Voice Activated Vehicle Signal Systems intention is to improve road safety by introducing a U-Turn signal in vehicles and activate turn indicator using voice command [1]. Work-in-Progress: Evaluating the performance of Voice Recognition Approaches for Autonomous Vehicular Systems evaluates two approaches for voice-controlled Arduino-based autonomous systems. The first approach uses an embedded voice recognition (VR) system that is interfaced with an Arduino-based hardware platform. In the second approach a speech recognition programming module is integrated into the development environment of the Arduino-based system [2]. Based on the research paper [3], the proposed steering mechanism was designed and the modelling was done in CATIA (V-5) and the same was analysed for static analysis for the proposed torque condition of the motor in ANSYS. All the results in the analysis were analysed for static analysis. The paper [4] suggests the concept and parameter design of a Robust Stair Climbing Compliant Modular Robot, capable of tackling stairs with overhangs. Along with establishing a concept design, robust design parameters were set to minimize performance variation. The Grey- based Taguchi Method was adopted or providing an optimal setting for the design parameters of the robot. The robot prototype was shown to have successfully scaled stairs of varying dimensions, with overhang, thus corroborating the analysis performed. An analysis method [5] to make the rocker bogie mechanism can climb up a stair was achieved in the work. The east coast of Malaysia faced a massive flood from heavy downpour, leading to huge flood. The flood carries the debris, soil and trees along their path, damaging the road and building structure, leaving the road become uneven. The research paper proposed an intelligent inclined motion control of an amphibious vehicle while moving on uneven terrain surface. The research paper [6] deals with the designing and modelling of stair climbing robot based on the well-known rocker bogie mechanism in rigid body dynamics module. The robots often suffer from undesired phenomenon slip, sticking and floating while climbing steps and stairs, which may cause instability of the mobile robot. The Taguchi method was used to chosen as an optimization tool to make trajectory of centre of mass close to straight line while all wheels keep in contact with ground during climbing stairs. In the Optimization, Seven kinematic parameters of rocker bogie mechanism were optimized which include four link lengths (l_1, l_2, l_3) and three wheel radius (R_1, R2, R3). The kinematic Model of proposed mechanism was built and it was simulated in ANSYS Rigid body dynamics. The result obtained shows the variation of centre of mass position with time, variation of velocity of joint with time, variation of force with time. In research paper [7], it was basically a suspension arrangement used in mechanical robotic vehicles used specifically for space exploration. The rocker-bogie suspension based rovers has been successfully introduced for the Mars Pathfinder and Mars Exploration Rover (MER) and Mars Science Laboratory (MSL) .The system favoured design for every space exploration company indulge in the business of space research. The motive of the research initiation was to understand mechanical design and its advantages of Rocker- bogie suspension system in order to find suitability to implement it in conventional loading vehicles to enhance their efficiency and also to cut down the maintenance related expenses of conventional suspension systems.

CHAPTER 3

PROPOSED SYSTEM

3.1 BLOCK DIAGRAM

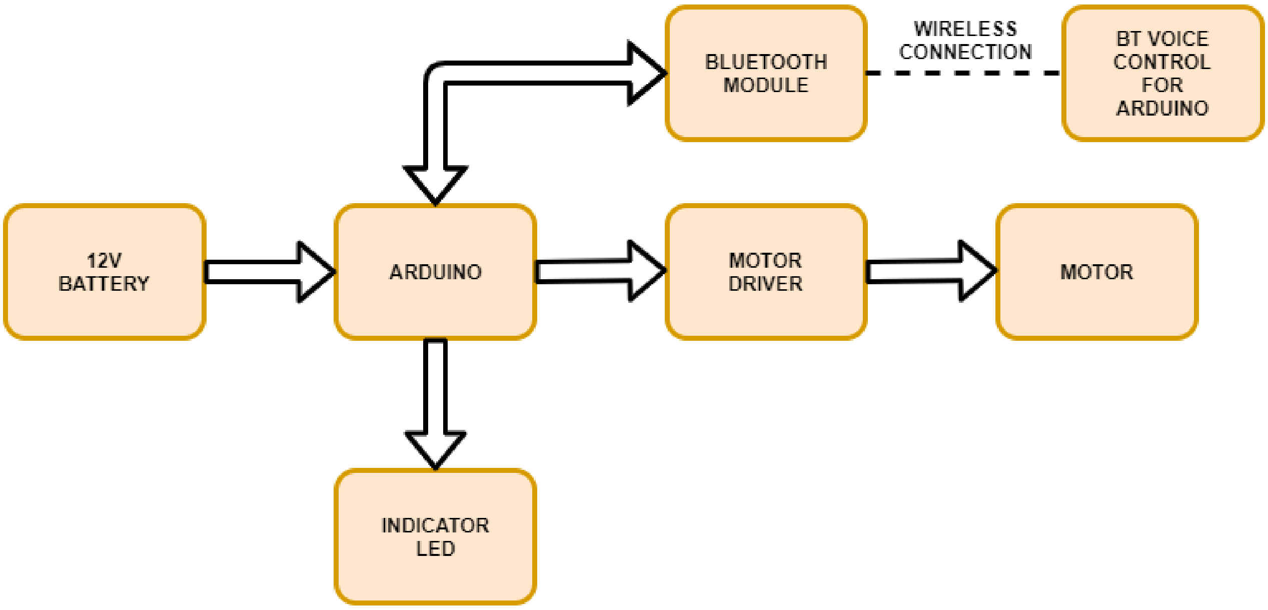


Fig 3.1 Block diagram

The rover chassis with rocker bogie mechanism forms the base of the electronic hardware components. The rover has a total of 6 wheels and each wheel is powered using 12v dc motors .Two individual wire connections are used to separately to interconnect the 3 wheels on each side of the rocker bogie mechanism. The rover employs Arduino Uno as its main microcontroller. Arduino Uno is a highly user friendly, open source coding platform and is coded using Arduino IDE. The whole hardware module of the project has been powered using a 12V Battery. On powering the module, the Arduino uno board is connected with Bluetooth module and Motor driver.

The Bluetooth module is used for interfacing the rover chassis with Bluetooth controlled voice recognition based mobile application. This mobile application has is used to give voice commands such as ‘FRONT’, ‘BACK’, ‘STOP’ which acts as instruction for the rover movement. 2 Indicator LED’s are also connected to the rover which glow in accordance with rover movement. When the rover is operated using the mobile app, the movement instruction is passed via Bluetooth module to Arduino Uno. It then passes the information on to the motor driver. The motor driver has been employed in order to reduce the complexity of the overall hardware connection and to decrease the lines of code of the software module. It also helps to easily interface the parallel wheels of the rocker bogie mechanism to the its two connecting ports. The motor driver then operates the motors of the wheel and thus the movement of robot can be observed.

3.2 CIRCUIT DIAGRAM

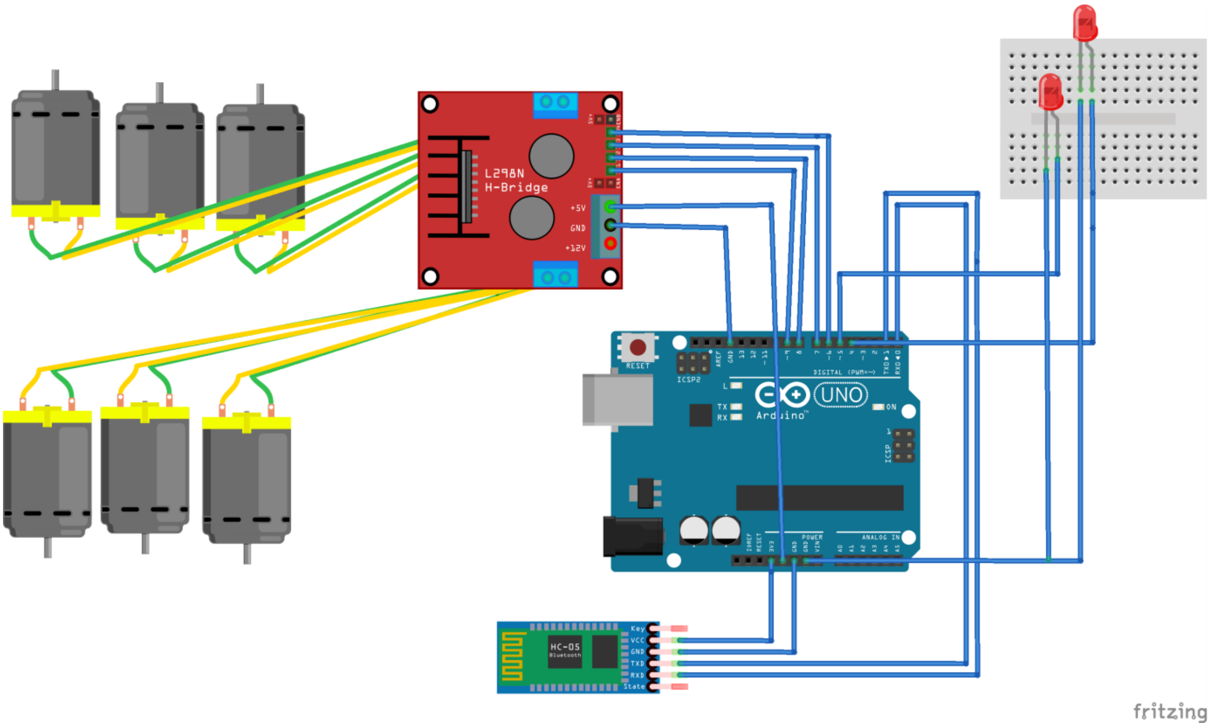


Fig 3.2 Circuit diagram

CHAPTER 4

HARDWARE USED

4.1 ARDUINO UNO BOARD

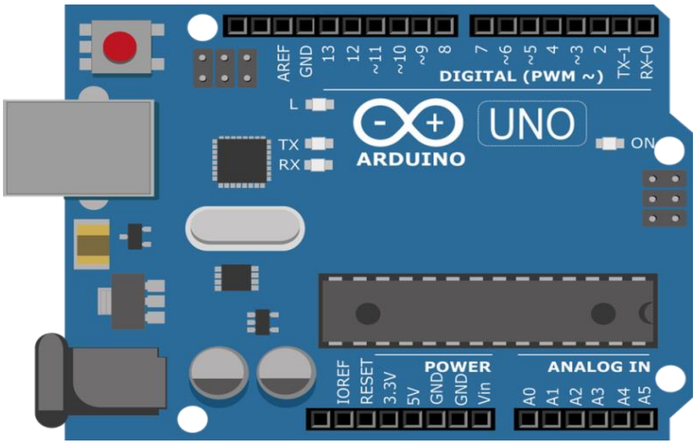


Fig 4.1 Arduino Uno Board

The Arduino UNO is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards, shields and other circuits. The board has 14 Digital pins, 6 Analog pins, and programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes pre-programmed with a boot loader that allows uploading new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.

The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2) programmed as a USB-to-serial converter.

4.1.1 GENERAL PIN FUNCTIONS OF ARDUINO UNO

- LED: There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- VIN: The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- 5V: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.
- 3.3V: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- GND: Ground pins.
- IOREF: This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.
- Reset: Typically used to add a reset button to shields which block the one on the board.

4.1.2 SPECIAL PIN FUNCTIONS OF ARDUINO UNO

Each of the 14 digital pins and 6 Analog pins on the Uno can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller .The Uno has 6 analog inputs, labelled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the analogReference() function.

In addition, some pins have specialized functions:

- Serial: pins 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: pins 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- PWM (Pulse Width Modulation) 3, 5, 6, 9, 10, and 11 Can provide 8-bit PWM output with the analogWrite() function.
- SPI (Serial Peripheral Interface): 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- TWI (Two Wire Interface): A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.
- AREF (Analog Reference): Reference voltage for the analog inputs

4.1.3 PINS OF ARDUINO UTILISED FOR INTERFACING

BLUETOOTH MODULE	3.3V, GND, PIN 0, PIN 1
MOTOR DRIVER	GND, PIN 6, PIN 7, PIN 8, PIN 9
12V BATTERY	GND

Table 4.1 Pins of Arduino utilised for interfacing

4.2 BLUETOOTH MODULE

Fig 4.2 Bluetooth module

Bluetooth is the specification for the use of low power radio communications to link phones, computers and other network devices over short distances without the use of wires. The general specifications are short range wireless connectivity, low power consumption, and automatic recognition. This module enables you to wirelessly transmit and receive serial data. It is a drop in replacement for wired serial connections allowing transparent two way data communication. Some of the popular Bluetooth apps are Bluetooth terminal, 8 Relay Bluetooth switch, Bluetooth app control robot, Bluetooth remote controller, RDL Bluetooth switch.

Some of its features are:

- 5V power operation
- UART interface
- Built in antenna
- 10 meters range
- Easy to use
- Minimal external components
- Status LEDs

4.2.1 USES OF BLUETOOTH MODULE

The HC-05 is a very cool module which can add two-way (full-duplex) wireless functionality to your projects. You can use this module to communicate between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality like a Phone or Laptop. There are many android applications that are already available which makes this process a lot easier. The module communicates with the help of USART at 9600 baud rate hence it is easy to interface with any microcontroller that supports USART. We can also configure the default values of the module by using the command mode. So if you looking for a Wireless module that could transfer data from your computer or mobile phone to microcontroller or vice versa then this module might be the right choice for you. However do not expect this module to transfer multimedia like photos or songs; you might have to look into the CSR8645 module for that.

4.2.2 PIN FUNCTIONS OF BLUETOOTH MODULE

PIN	PIN NAME	DESCRIPTION
1	Enable/Key	This pin is used to toggle between Data Mode(Set low) and AT command mode(Set high).By default it is in Data mode.
2	Vcc	Powers the module connect to +5v Supply voltage
3	Ground	Ground pin of module, connect to system ground
4	TX Transmitter	Transmits Serial data .Everything received via Bluetooth will be given out by this pin as serial data
5	RX Receiver	Receives serial data, every serial data given to this pin will be broadcasted via Bluetooth
6	State	The state pin is connected to on board LED, it can be used as a feedback to check if Bluetooth is working properly.
7	LED	Blink once in 2s; Module has entered command mode. Repeated blinking; Waiting for connection in data mode. Blink twice in 1s;Connection successful in data mode.
8	Button	Used to control the key enable pin

Table 4.2 Pin functions of Bluetooth module

4.2.3 PINS UTILISED FOR INTERFACING

The RXD and TXD pins of the Bluetooth module is connected to the TXD and RXD of the Arduino board respectively. The VCC and GND are connected to 3V and GND of the Arduino board.

BLUETOOTH MODULE	ARDUINO UNO
RXD	TXD
TXD	RXD
VCC	3V
GND	GND

Table 4.3 Pins utilised for interfacing Bluetooth module

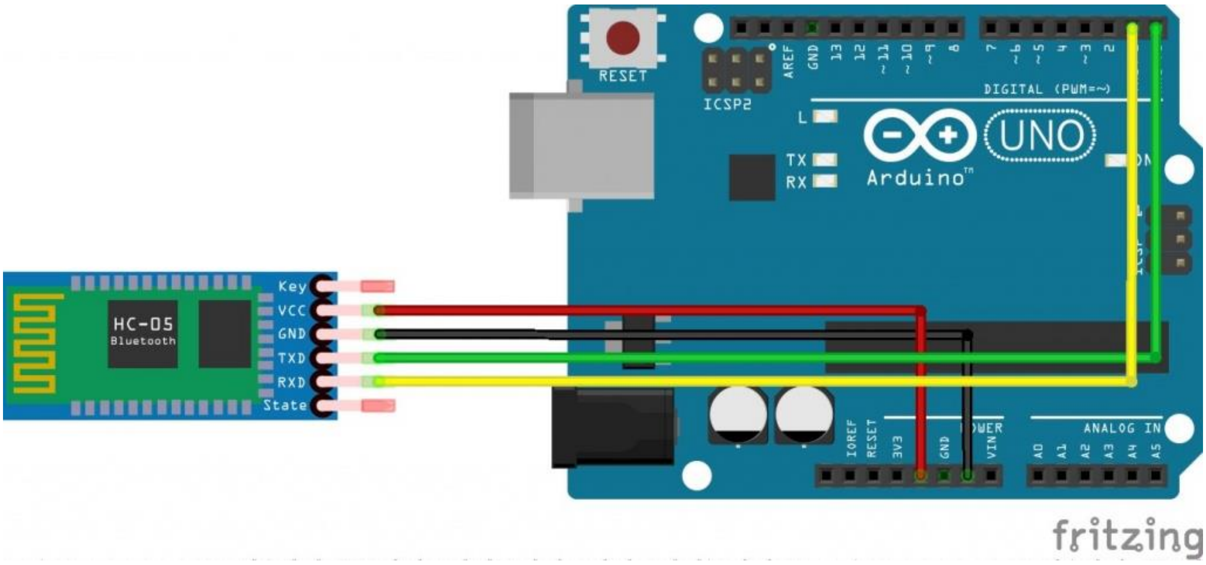


Fig 4.3 Interfacing Bluetooth module with Arduino Uno

4.3 12V BATTERY

12v Battery is a device used to provide power to electrical devices. It consists of one or more electrochemical cells. It converts chemical energy directly into electrical energy. There are generally two types – primary and secondary batteries. Primary batteries are used once and discarded. Secondary batteries can be discharged and recharged multiple times using by an applied electric current. A 12 volt battery is available as a non-rechargeable alkaline battery or in rechargeable forms. This battery is often used in various outdoor applications that require greater amounts of energy in order to operate as desired. However, it may be recommended to use a rechargeable battery if the electronic item is one that is used regularly. Non-rechargeable batteries can become quite expensive if they must be purchased regularly. One of the most common uses of a 12 volt battery is for transportation applications, such as in cars and boats. In these cases, the battery may be able to be recharged as current is only needed to start the vehicle. After that, the alternator takes over and runs the electrical system, if it is functioning properly. The alternator also puts a charge back into the battery. The sizes of 12 volt batteries vary widely based on the amp hours they are designed to produce. They can be very heavy and large or relatively small also

Fig 4.4 12V Battery

4.4 MOTOR DRIVER

The L298N is an integrated monolithic circuit in a 15- lead Multiwatt and PowerSO20 packages. It is a high voltage , high current dual full-bridge driver de-signed to accept standard TTL logic level sand drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the in-put signals . The emitters of the lower transistors of each bridge are connected together rand the corresponding external terminal can be used for the connection of an external sensing resistor. An additional Supply input is provided so that the logic works at a lower voltage.

The few areas where L298N is preferred:

1. L298N is basically used where H- BRIDGE is required.
2. Where a high power motor driver is required. In the marked there are H-bridges like L293N which are used for low powered application while L298N is specially designed for high power applications.
3. Where current control and PWM operable single chip device is needed.
4. The chip is preferred when control unit can only provide TTL outputs

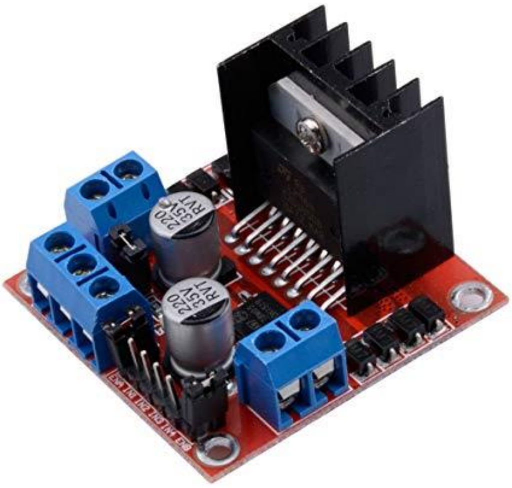


Fig 4.5 Motor driver

Features and Specifications

- Operating voltage range: +5 to +46V
- Maximum supply voltage:50V
- Maximum Input and Enable Voltage:+7V
- Maximum current allowed to draw through each output: 3A
- TTL control inputs
- Total power dissipation:25W
- Operating temperature: -23°C to 130°C
- Storage Temperature: -40°C to 150°C.

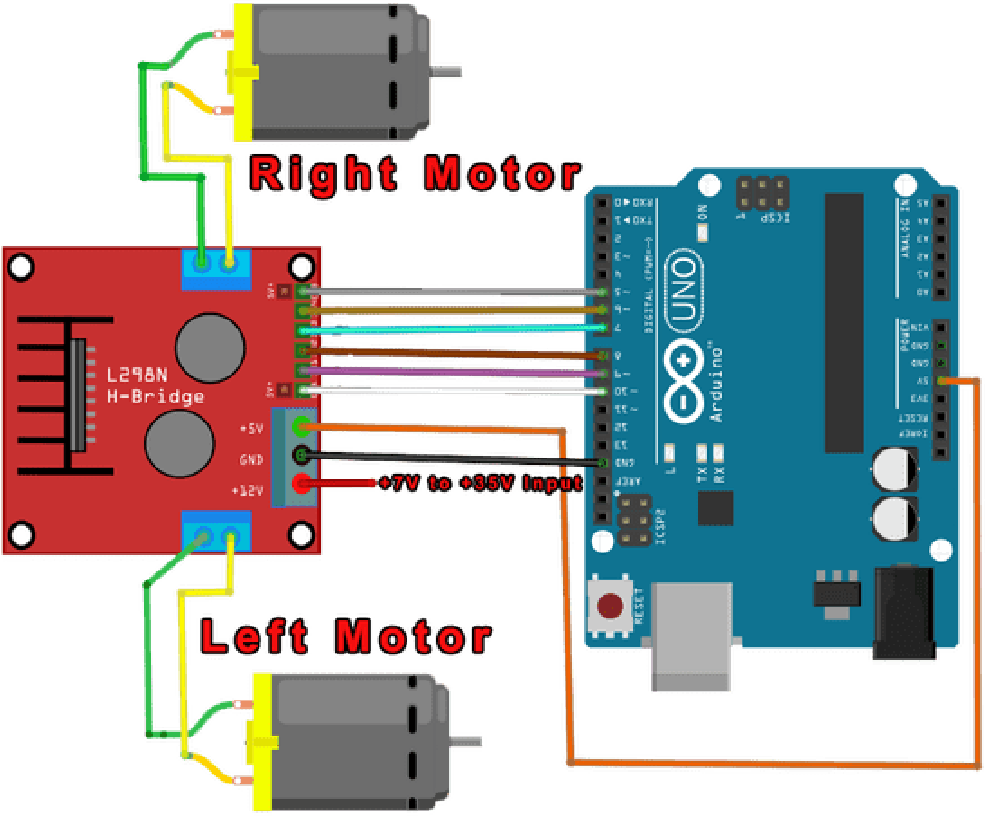


Fig 4.6 Interfacing motor driver to Arduino Uno

4.4.1 PINS FUNCTIONS OF MOTOR DRIVER

PIN	NAME	DESCRIPTION
1	SENSE A	The sense resistor is placed between this pin and ground to control the current of the load.
15	SENSE B	The sense resistor is placed between this pin and ground to control the current of the load.
3	OUT 2	Outputs of the Bridge A
4	VS	Supply voltage for the Power Output Stages.
5	INPUT 1	TTL compatible inputs of the Bridge A
7	INPUT 2	TTL compatible inputs of the Bridge A
6	ENABLE A	TTL compatible Enable Input: the L state disables the bridge A(enable A) and/or the bridge B(enable B)
11	ENABLE B	TTL compatible Enable Input: the L state disables the bridge A(enable A) and/or the bridge B(enable B)
8	GND	It acts as ground pin for the circuit.
9	VSS	Supply voltage for the Logic Blocks.
10	INPUT 3	TTL compatible inputs of the bridge B.
12	INPUT 4	TTL compatible inputs of the bridge B.
13	OUTPUT 3	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15
14	OUTPUT 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15

Table 4.4 Pin functions of motor driver

Applications

- Robotic arms
- Robots
- Relay drivers
- Industrial machines.
- Engineering systems.
- Measuring instruments

4.4.2 PINS UTILISED FOR INTERFACING

The IN1, IN2, IN3, IN4 of the motor driver is connected to PIN6, PIN7, PIN8 and PIN9 of the arduino board respectively.

L298N MOTOR DRIVER	ARDUINO UNO
IN1	PIN6
IN2	PIN7
IN3	PIN8
IN4	PIN9
5V	5V
GND	GND

Table 4.5 Pins utilised for interfacing motor driver

The Positive terminals of motors A, B, C are connected to OUT 1 of the Motor driver. The Negative terminals of motors A, B, C are connected to OUT 2 of the Motor driver. The Positive terminals of motors D, E, F are connected to OUT 3 of the Motor driver. The Negative terminals of motors D, E, F are connected to OUT 4 of the Motor driver.

L298N MOTOR DRIVER	MOTORS
OUT 1	+VE MOTOR A,B,C
OUT 2	-VE MOTOR A,B,C
OUT 3	+VE MOTOR D,E,F
OUT 4	-VE MOTOR D,E,F

Table 4.6 Pins utilised for interfacing motors with motor driver

The 12V pin of the Motor driver is connected to the positive terminal of the Battery. The GND pin of the Motor driver is connected to the negative terminal of the Battery.

L298N MOTOR DRIVER	BATTERY
12V	+VE
GND	-VE

Table 4.7 Pins utilised for interfacing motor driver to battery

4.5 12V DC MOTOR

A DC motor is any of a class of rotary electrical machines that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current flow in part of the motor. DC motors were the first form of motor widely used, as they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances where as the universal motors can operate on direct current . Larger DC motors are currently used in propulsion of electric vehicles, elevator and hoists, and in drives for steel rolling mills. The advent of power electronics has made replacement of DC motors with AC motors possible in many applications. DC Motor directly provides rotary motion and coupled with wheels, drums or cables. It is widely used as prime movers in the industry today.

Fig 4.7 12V DC Motor

4.6 JUMP WIRES

A jump wire s an electrical wire, or group of them in a cable, with a connector or pin at each end which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment. Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with breadboards and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. In fact, it doesn’t get much more basic than jumper wires. Though jumper wires come in a variety of colours, the colours don’t actually mean anything. This means that a red jumper wire is technically the same as a black one. But the colours can be used to your advantage in order to differentiate between types of connections, such as ground or power. Though jumper wires come in a variety of colours, the colours don’t actually mean anything. This means that a red jumper wire is technically the same as a black one. But the colours can be used to your advantage in order to differentiate between types of connections, such as ground or power.

Fig 4.8 Jump wires

4.7 LIGHT EMITTING DIODES(LED)

A light-emitting diode (LED) is a semiconductor device that emits light when an electric current is passed through it. Light is produced when the particles that carry the current (known as electrons and holes) combine together within the semiconductor material. Since light is generated within the solid semiconductor material, LEDs are described as solid-state devices. The term solid-state lighting, which also encompasses organic LEDs (OLEDs), distinguishes this lighting technology from other sources that use heated filaments (incandescent and tungsten halogen lamps) or gas discharge (fluorescent lamps). LEDs are comprised of compound semiconductor materials, which are made up of elements from group III and group V of the periodic table (these are known as III-V materials).

The main semiconductor materials used to manufacture LEDs are:

- Indium gallium nitride (InGaN): blue, green and ultraviolet high-brightness LEDs
- Aluminum gallium indium phosphide (AlGaInP): yellow, orange and red high-brightness LEDs
- Aluminum gallium arsenide (AlGaAs): red and infrared LEDs
- Gallium phosphide (GaP): yellow and green LEDs

Fig 4.9 LED

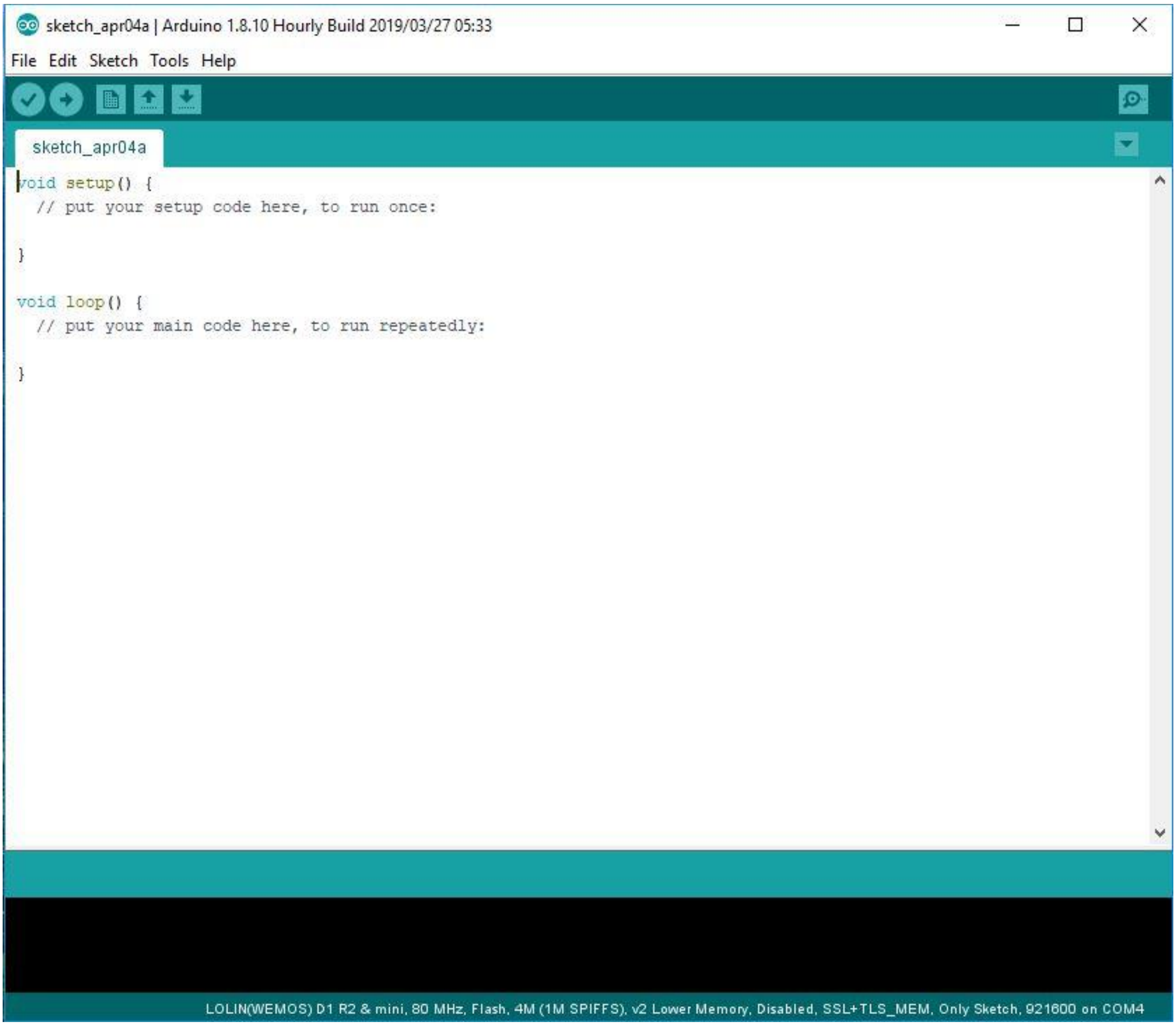
CHAPTER 5

SOFTWARE USED

5.1 ARDUINO IDE

Fig 5.1 Arduino Logo

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.



sketch_apr04a | Arduino 1.8.10 Hourly Build 2019/03/27 05:33

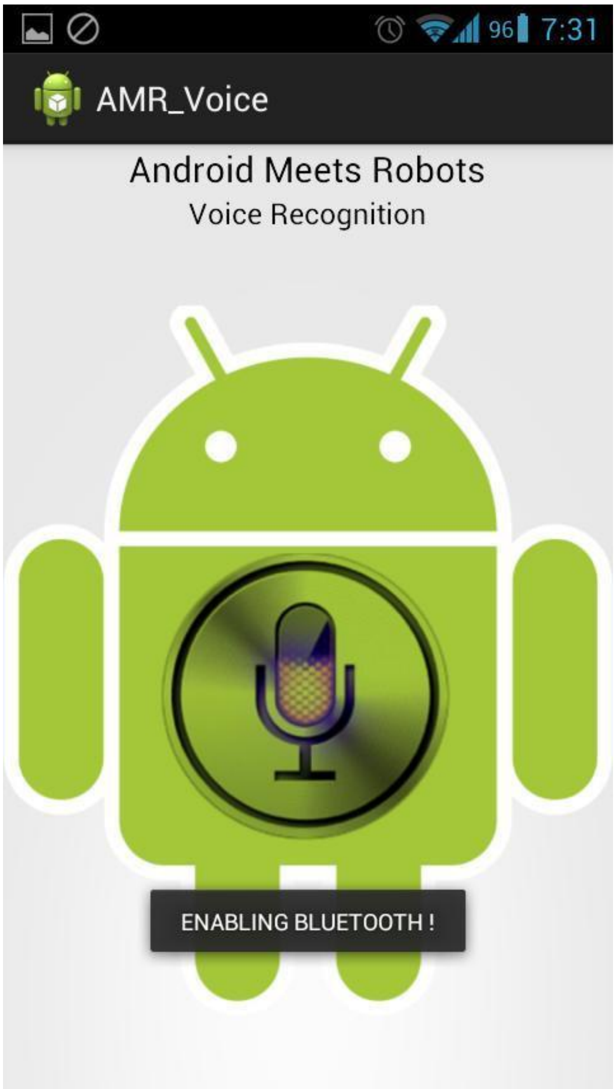
```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

LOLIN(WEMOS) D1 R2 & mini, 80 MHz, Flash, 4M (1M SPIFFS), v2 Lower Memory, Disabled, SSL+TLS_MEM, Only Sketch, 921600 on COM4

Fig 5.2 Arduino IDE

5.2 BT VOICE CONTROL FOR ARDUINO

BT voice control for Arduino is an android application which is used for voice recognition. It employs android mobiles internal voice recognition to pass voice commands to your robot. Pairs with Bluetooth Serial Modules and sends in the recognized voice as a string for example if you say Hello the android phone will return a sting *Hello# to your Bluetooth module *and # indicate the start and stop bits. Thus it can be used with any micro controller which can handle strings Examples Platforms: Arduino, ARM, PICAXE, MSP43 , 8051 based and many other processors and controllers.



ENABLING BLUETOOTH !

Fig 5.3 BT Bluetooth control interface

CHAPTER 6

EXPERIMENTAL OUTPUT

6.1 EXPERIMENTAL SETUP OF THE ROVER CHASSIS

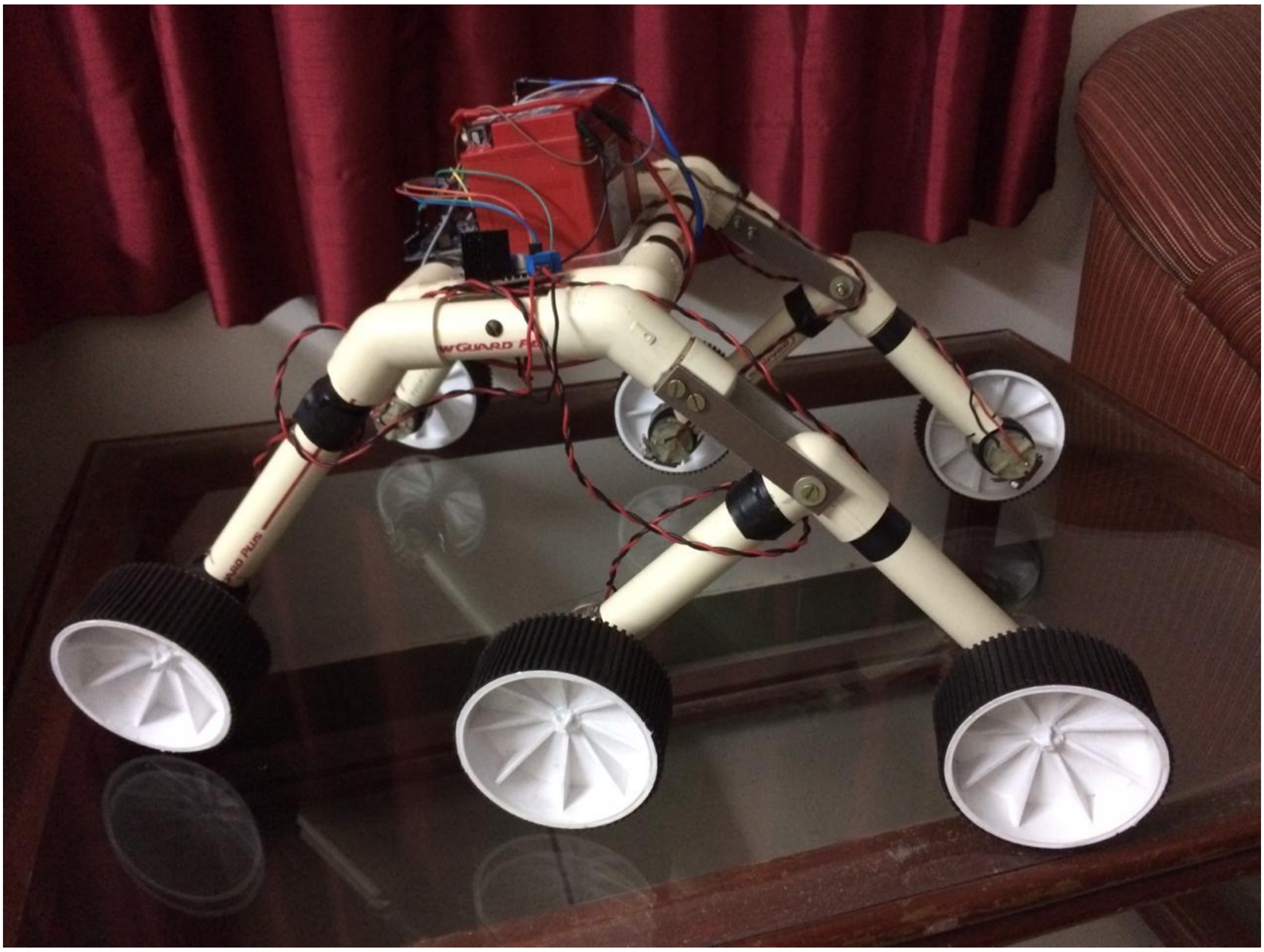


Fig 6.1 Experimental setup of the rover chassis

6.2 EXPERIMENTAL OUTPUT OF THE ROVER MOVEMENT

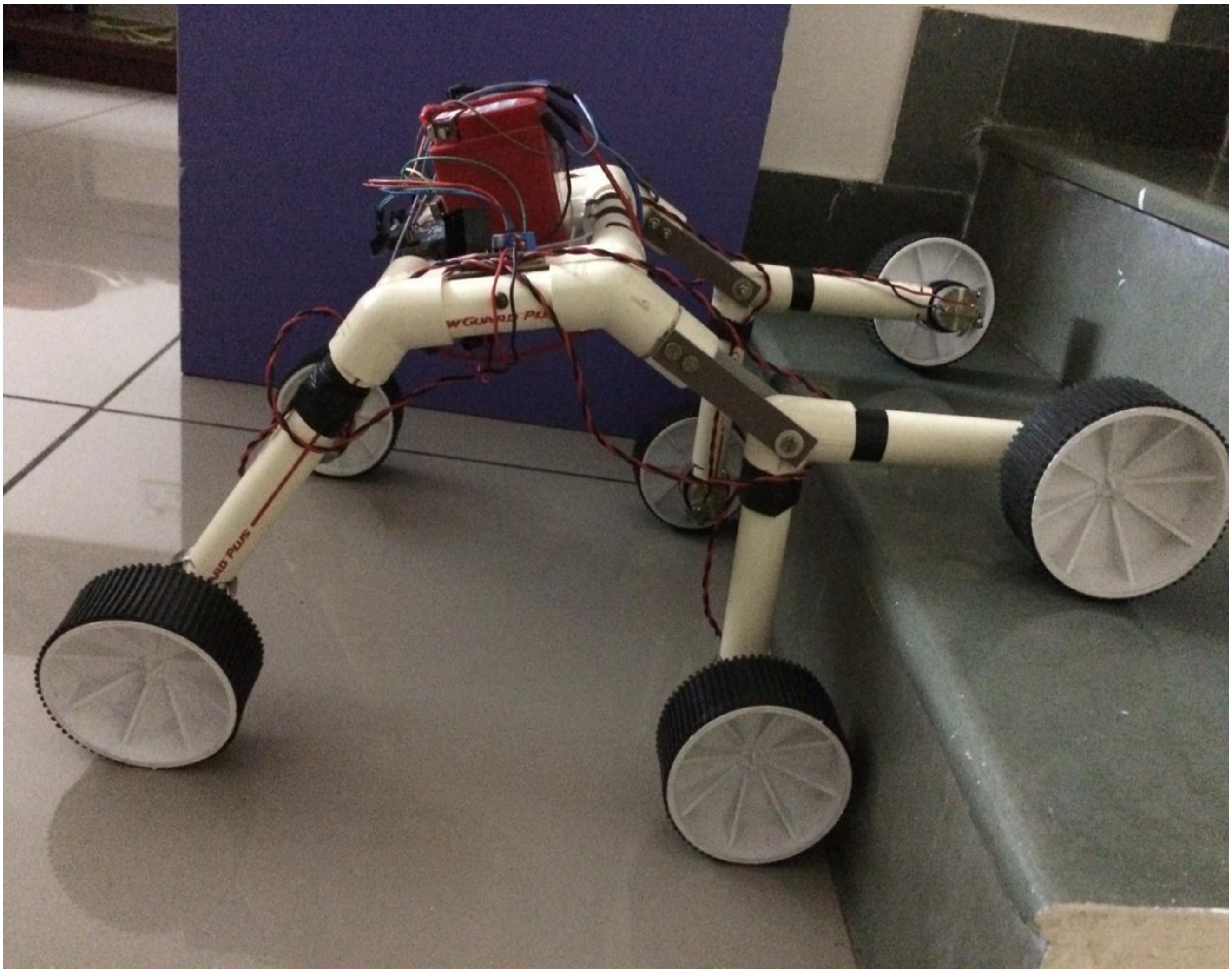


Fig 6.2 Experimental output of the rover movement

CHAPTER 7

FUTURE SCOPE AND CONCLUSION

7.1 FUTURE SCOPE

- Different types of sensors such as the Temperature sensors, Humidity sensors, and Pressure sensors can be attached to monitor the climatic changes and analyse the atmospheric data.
- The mechanism of Rocker-Bogie and can be employed for times of calamity. During calamities, the rover with robotic arm can help in saving lives.
- The rovers can also be developed into a wheelchair for the physically challenged. The mechanism of the Rocker-Bogie can be developed, so that it can also climb the stairs. Since voice recognition is used for wireless communication, it will be easier for everyone to use.
- Another most important application where this mechanism can be used is in Agriculture. It is designed to minimize the labour of farmers in addition to increasing the speed and accuracy of the work. The rover can be employed with image sensing so that it can help the farmers in identifying the different types of diseases in plants.
- It can be used as assistive robots for people with disabilities or in industrial applications such as work robots.
- Indoor assistive robots which will navigate around to pick up objects from one place and place them at another using speech commands technique.
- Surveillance applications to send live feed from camera and track down an objects.

7.2 CONCLUSION

The Mars rover exploration robot designed is a potential space exploration robot which can be sent to space to monitor the various important atmospheric parameters is an alien planet. It is a six wheeled rover which is based on the rocker bogie mechanism. This mechanism consists of a rocker and a bogie pivoted to the middle of the rover chassis. It forms the backbone of the rover and helps it to easily move over uneven terrain. This arrangement helps the rover to surmount any obstacle whole size is lesser than twice the diameter of the rovers wheel. The rocker bogie system is a relatively slow moving arrangement with powerful load carrying capacity and high reliability. Apart from this system the it has been powered by Arduino Uno. It forms the main microcontroller and communicates with Bluetooth module and Motor driver of the rovers. This Bluetooth module is in turn connected wirelessly to mobile application and acts as receiver. It wirelessly receives voice commands from the mobile application which helps to move the rover. The movement of the rover can be easily controlled using this mobile application. Along with the rover movement automates indicator led system is also attached which glows in accordance with the direction of the rover. Thus a fully functional Rocker bogie mechanism based voice-controlled rover with turn indicator has been implemented using Arduino.

REFERENCES

[1] Jewel L. Dohan, “Voice Recognition/Voice Activated Vehicle Signal System”, Patent Appl. No: 12/806,929, pp 8-10, Aug.24 2010

[2] Shonda L. Bernadin and Rahul Patel, Earl Smith, “Work-in-Progress: Evaluating the performance of Voice Recognition Approaches for Autonomous Vehicular Systems”, Southeast Con 2015, April 9 - 12, 2015

[3]P. Panigrahi, A. Barik, Rajneesh R. & R. K. Sahu, “Introduction of Mechanical Gear Type Steering Mechanism to Rocker Bogie”, Imperial Journal of Interdisciplinary Research (IJIR) Vol-2, Issue-5, ISSN: 2454-1362, 2016.

[4]A. Bhole, S. H. Turlapati, Raja shekhar V. S, J. Dixit, S. V. Shah, Madhava Krishna K, “Design of a Robust Stair Climbing Compliant Modular Robot to Tackle Overhang on Stairs” ArXiv: 1607.03077v1 [cs.RO], 11 Jul 2016.

[5]M. D. Manik, A.S. Chauhan, S.Chakraborty, V. R.Tiwari, “Experimental Analysis of climbing stairs with the rocker-bogie mechanism”, Vol-2 Issue-2 P.No. 957-960IJARIIE-ISSN (O)-2395-4396, 2016.

[6] Y.L. Maske, S.V. Patil, S.Deshmukh, “Modelling and MBD simulation of stair climbing robot with rocker bogie Mechanism”, International Journal of Innovative Research in Technology, 101743, Volume 1 Issue 12, Page no. 267-273, ISSN: 2349-6002, 2015.

[7]N.Yadav, B. Bhardwaj, S. Bhardwaj, “Design analysis of Rocker Bogie Suspension System and Access the possibility to implement in Front Loading Vehicles”, IOSR Journal of Mechanical and Civil Engineering.

APPENDIX:

```
#include <AFMotor.h>
#include <Servo.h>
String voice;
AF_DCMotor motor1 (1, MOTOR12_1KHZ);
AF_DCMotor motor2 (2, MOTOR12_1KHZ);
Servo myServo;
int LED1 = A0;
int LED2 = A1;
int buzzerPin = A2;
void setup()
{
  Serial.begin(9600);
  myServo.attach(10);
  myServo.write(90);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
}
void loop()
{
  while (Serial.available()){
     delay(10);
    char c = Serial.read();
    if (c == '#') {break;}
    voice += c;
  }
  if (voice.length() > 0){
    if(voice == "*go ahead"){
      forward_car();      }
    else if(voice == "*go back"){
      back_car();
      }
    else if(voice == "*turn right") {
      right_car();}
    else if(voice == "*turn left") {
      left_car();
      digitalWrite(LED1,HIGH);
    }
    else if(voice == "*turn on light") {
      digitalWrite(LED2,HIGH);
      LED_on();
    }
    else if(voice == "*turn off light") {
      LED_off();
    }
    else if(voice == "*buzzer") {
      buzzer_on();
    }
    else if(voice == "*stop") {
      stop_car();
    }
      voice="";   }}
void forward_car()
{
  motor1.run(FORWARD);
  motor1.setSpeed(170);
```

```
 motor2.run(FORWARD);
 motor2.setSpeed(170);
 delay(2000);
 motor1.run(RELEASE);
 motor2.run(RELEASE);}
 void back_car()
{
 motor1.run(BACKWARD);
 motor1.setSpeed(170);
 motor2.run(BACKWARD);
 motor2.setSpeed(170);
 delay(2000);
 motor1.run(RELEASE);
 motor2.run(RELEASE);
}
 void right_car()
{
 myServo.write(0);
 delay(1000);
 myServo.write(90);
 delay(1000);
 motor1.run(FORWARD);
 motor1.setSpeed(170);
 motor2.run(BACKWARD);
 motor2.setSpeed(170);
 delay(1000);
 motor1.run(RELEASE);
 motor2.run(RELEASE);
}
```

```
  void left_car()
{
  myServo.write(180);
  delay(1000);
  myServo.write(90);
  delay(1000);
  motor1.run(BACKWARD);
  motor1.setSpeed(170);
  motor2.run(FORWARD);
  motor2.setSpeed(170);
  delay(1000);
  motor1.run(RELEASE);
  motor2.run(RELEASE);}
void LED_on ()
{
  digitalWrite(LED1, HIGH);
  digitalWrite(LED2, HIGH);}
void LED_off ()
{
  digitalWrite(LED1, LOW);
  digitalWrite(LED2, LOW);}
void buzzer_on ()
{tone(buzzerPin, 100);
  delay(800);
  noTone(buzzerPin);}
void stop_car ()
{motor1.run(RELEASE);
motor2.run(RELEASE);}
```