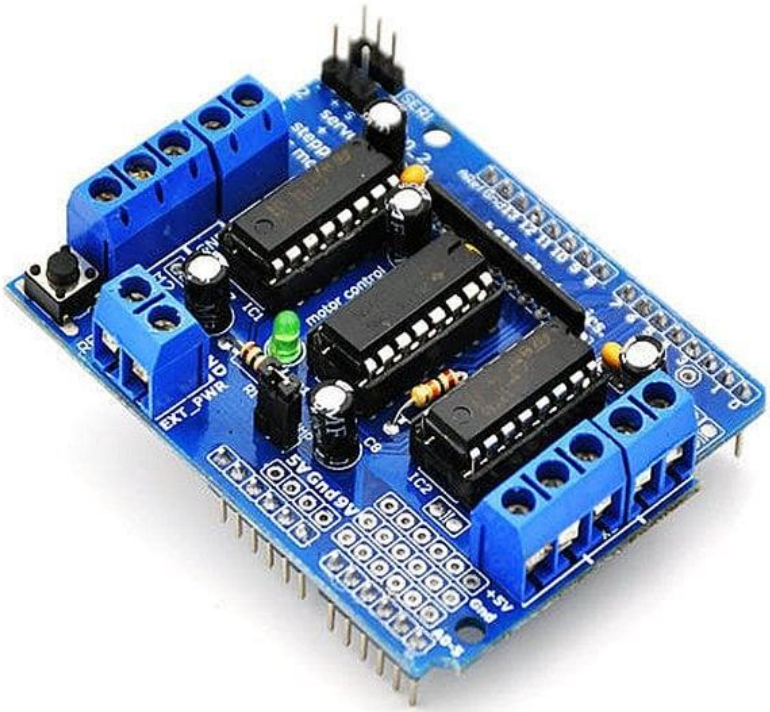
INTRODUCTION

The origins of robotic development may be traced back to a few fundamental concepts. It reduces human work and may be used in a variety of domains, including military, surveillance, and industrial picking. Wireless technology is also being used to construct robotic automobiles. We can link a larger number of Robotic Cars for wars. Self driven cars are also part of new era. Self driven cars are bigger scale of my mini project obstacle avoiding car .

Components

1.L293D ADUINO MOTOR DRIVER SHIELD



SunFounder L293D is a monolithic integrated, high voltage, high current, 4-channel driver. Basically this means using this chip you can use DC motors and power supplies of up to 16 Volts, thats some pretty big motors and the chip can supply a maximum current of 600mA per channel, the L293D chip is also what's known as a type of H-Bridge. The H-Bridge is typically an electrical circuit that enables a voltage to be applied across a load in either direction to an output.

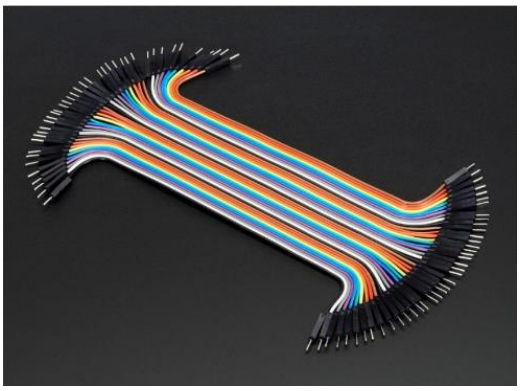
2. Jumper Wire

A jumper wire is an electrical wire (or a group of them in a cable) with a connector or

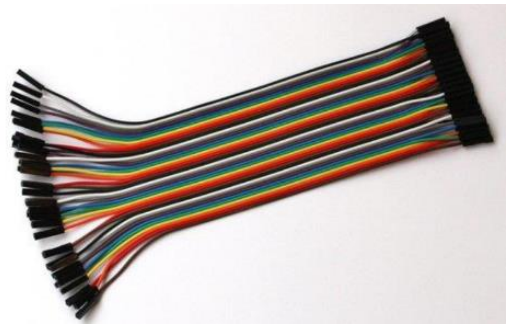
pin at each end (or sometimes without – simply "tinned") that is used to connect the

components of a breadboard or other prototype or test circuit internally or with other

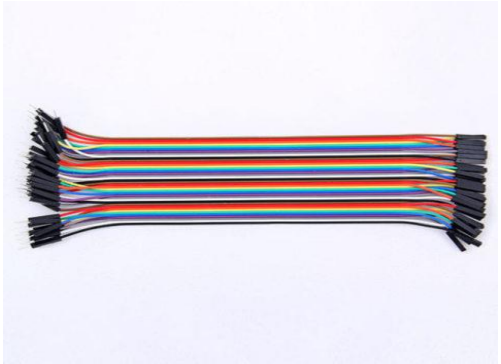
equipment or components without soldering.



M2M



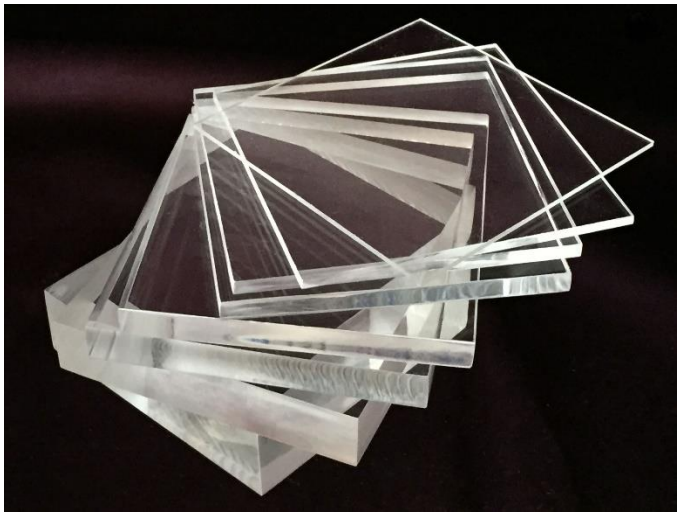
F2F



M2F

3. Car Robot Kit

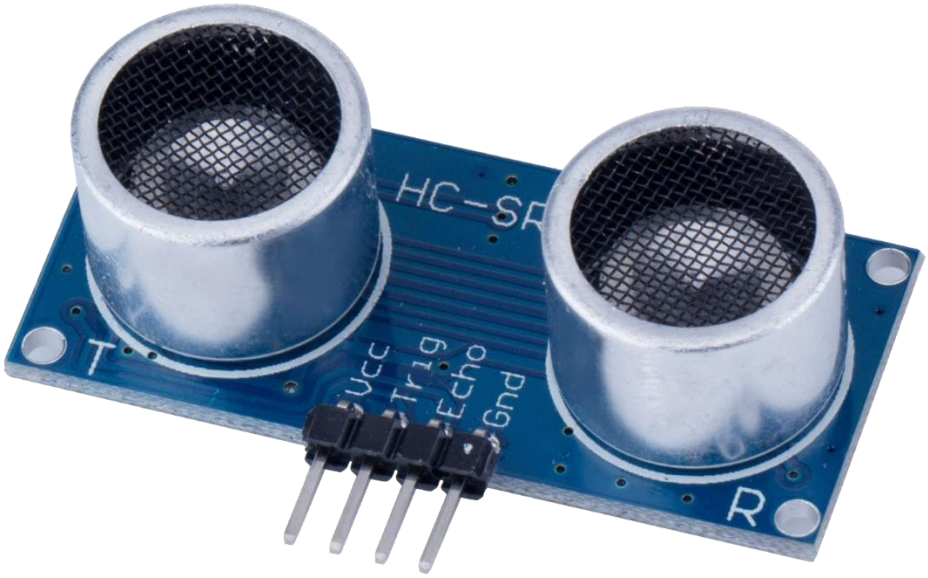
A robot kit is a unique construction kit designed for the construction of robots,particularly autonomous mobile robots. This is a 4-wheel double-layer Robot vehicle chassis kit that you may build yourself. The four sets of Geared Motors and Wheels are included. The transparent chassis included in this kit allows for dynamic handling of the components installed on your robotic vehicle. (I have used a transaparent acrylic sheet instead of double layer .)



4. 3.7V 2600 mAh Li ion Battery

A lithium-ion polymer (LiPo) battery is a rechargeable Li-ion battery that uses a polymer electrolyte rather than the liquid electrolyte used in traditional Li-ion batteries. LiPo chemistries come in a number of forms. The electrolyte in all of them is a high conductivity gel polymer. LiPo batteries have greater specific energies than ordinary lithium batteries and are frequently employed in systems where weight is an issue, such as mobile devices, drones, and some electric vehicles.

5.ULTRASONIC SENSOR



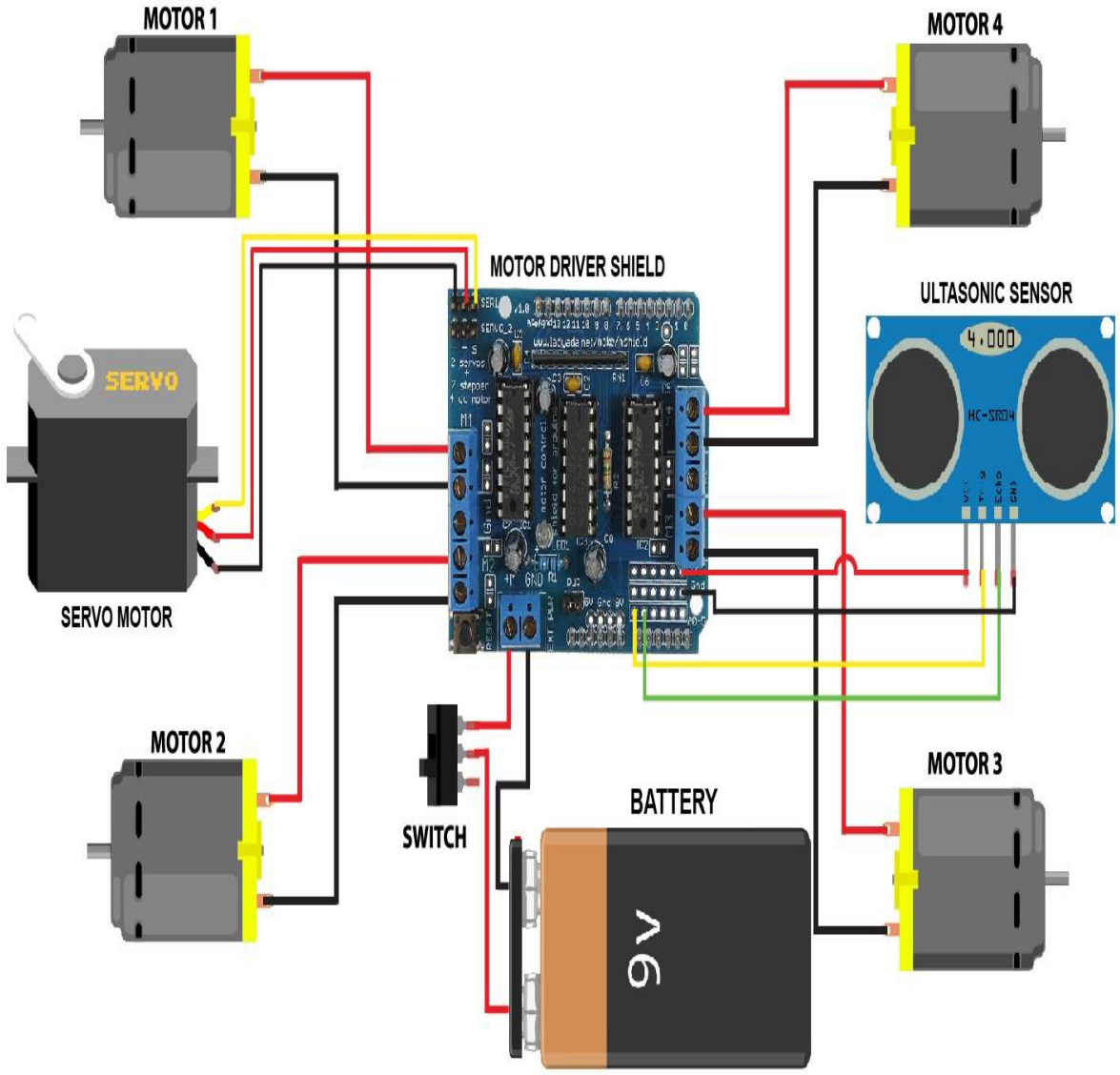
Ultrasonic Sensors also known as transceivers when they both send and receive work on a principle similar to radar or sonar which evaluate attributes of a target by interpreting the echoes from radio or sound waves respectively. Ultrasonic sensors generate high frequency sound waves and evaluate the echo which is received back by the sensor. Sensors calculate the time interval between sending the signal and receiving the echo to determine the distance to an object.

6. SERVO

Servos (also **RC servos**) are small, cheap, mass-produced servomoters or other actuators used for radio control and small-scale robotics.

Most servos are rotart actuators although other types are available. Linear actuators are sometimes used, although it is more common to use a rotary actuator with a bellcrank and pushrod. Some types, originally used as sail winches for model yatching can rotate continuously.

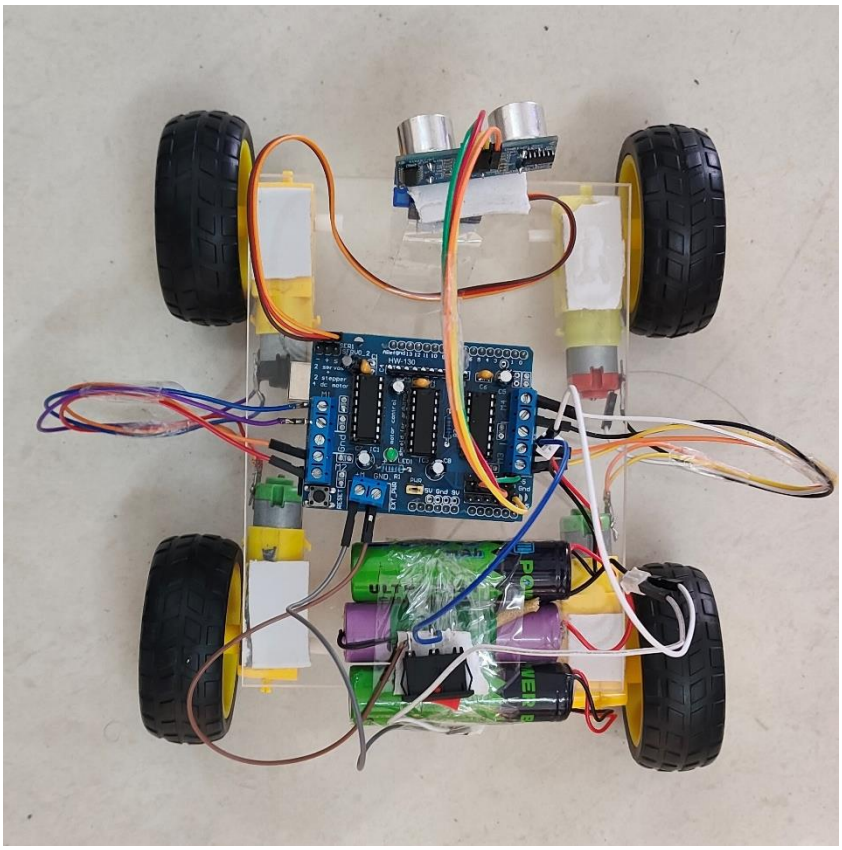
CIRCUIT DIAGRAM



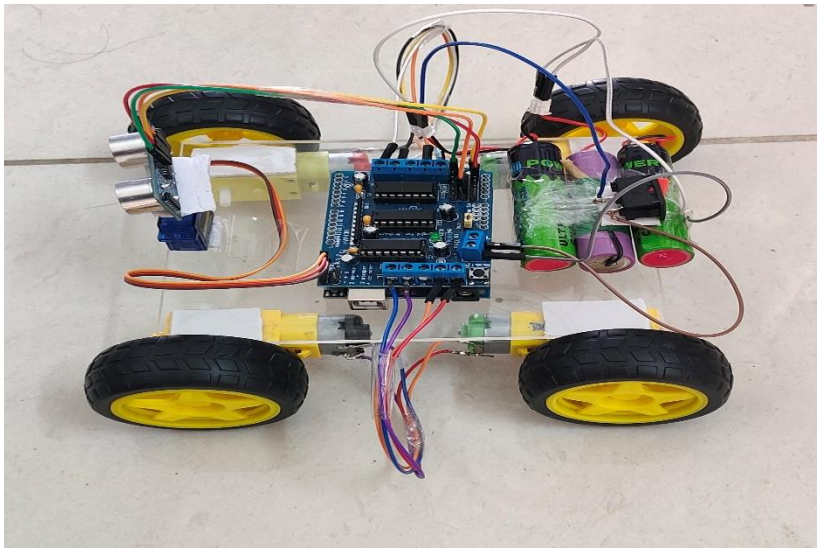
Connenct the components as shown in the figure.

PICTURES

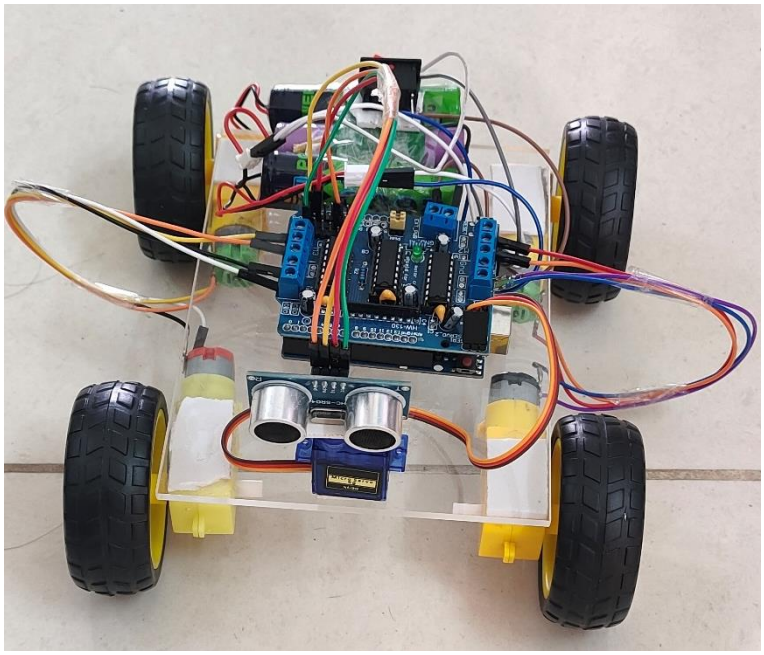
TOP VIEW



SIDE VIEW



FRONT VIEW



CODE ANALYSIS

```
//ARDUINO OBSTACLE AVOIDING CAR//


/* Name - Rachakonda Eshwar
 *  Roll - 2101AI25
 *  Email- eshwar_2101ai25@iitp.ac.in
 */
// Before uploading the code include the necessary libraries//
//AFMotor Library
//NewPing Library
//Servo Library
// To Install the libraries go to sketch >> Include Library >> Add .ZIP File >>
//Select the Downloaded ZIP files//


#include <AFMotor.h>
#include <NewPing.h>
#include <Servo.h>

#define TRIG_PIN A4
#define ECHO_PIN A5
#define MAX_DISTANCE 200
#define MAX_SPEED 190 // sets speed of DC  motors
#define MAX_SPEED_OFFSET 20

NewPing sonar(TRIG_PIN, ECHO_PIN, MAX_DISTANCE);

AF_DCMotor motor1(1, MOTOR12_1KHZ);
AF_DCMotor motor2(2, MOTOR12_1KHZ);
AF_DCMotor motor3(3, MOTOR34_1KHZ);
AF_DCMotor motor4(4, MOTOR34_1KHZ);
Servo myservo;
```

To start with , the Pins that connect motor driver to Arduino board have been specified . Then there are the NewPing , AFMotor , Servo libraries . The variables were then established to store values of the ping.

```
void setup() {

  myservo.attach(10);
  myservo.write(65);
  delay(2000);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
}
```

Setup servo values to stabilise the angle . Calculate distanes at an interval of 100ms .

```
void loop() {
 int distanceR = 0;
 int distanceL =  0;
 delay(40);

 if(distance<=30)
 {
  moveStop();
  delay(300);
  moveBackward();
  delay(100);
  moveStop();
  delay(300);
  distanceR = lookRight();
  delay(200);
  distanceL = lookLeft();
  delay(200);

   if(distanceR>=distanceL)
  {
    turnRight();
    moveStop();
  }else
  {

    turnLeft();

    moveStop();
  }
 }else
 {
  moveForward();
 }
 distance = readPing();
}
```

Initiate loop for calculating distances in left and right directions with delays in the between them . Some methods are used which will implement the actions required after reading the distances .

```
int lookRight()
{
    myservo.write(0);
    delay(500);
    int distance = readPing();
    delay(100);
    myservo.write(65);
    return distance;
}

int lookLeft()
{
    myservo.write(130);
    delay(500);
    int distance = readPing();
    delay(100);
    myservo.write(65);
    return distance;
    delay(100);
}

int readPing() {
  delay(70);
  int cm = sonar.ping_cm();
  if(cm==0)
  {
    cm = 250;
  }
  return cm;
```

Look right and look left functions . Central angle is 65 . 0 degree is look right side angle. It rotates 65 degrees in clock and anti clock wise direction .

```
void moveStop() {
  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);
  }

void moveForward() {

 if(!goesForward)
  {
    goesForward=true;
    motor1.run(FORWARD);
    motor2.run(FORWARD);
    motor3.run(FORWARD);
    motor4.run(FORWARD);
   for (speedSet = 0; speedSet < MAX_SPEED; speedSet +=2) // slowly bring the speed up to avoid loading down the batteries too quickly
   {
    motor1.setSpeed(speedSet);
    motor2.setSpeed(speedSet);
    motor3.setSpeed(speedSet);
    motor4.setSpeed(speedSet);
    delay(5);
   }
  }
}
```

Stopping upon detection of obstacle infront of it .

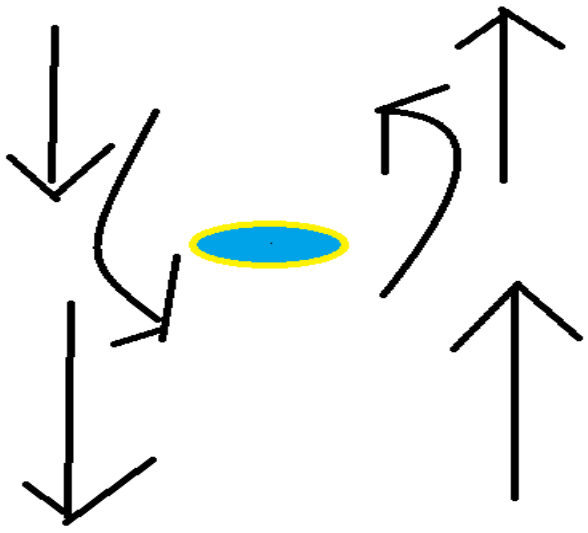
```
void moveBackward() {
    goesForward=false;
    motor1.run(BACKWARD);
    motor2.run(BACKWARD);
    motor3.run(BACKWARD);
    motor4.run(BACKWARD);
  for (speedSet = 0; speedSet < MAX_SPEED; speedSet +=2) // slowly bring the speed up to avoid loading down the batteries too quickly
  {
    motor1.setSpeed(speedSet);
    motor2.setSpeed(speedSet);
    motor3.setSpeed(speedSet);
    motor4.setSpeed(speedSet);
    delay(5);
  }
}

void turnRight() {
  motor1.run(FORWARD);
  motor2.run(FORWARD);
  motor3.run(BACKWARD);
  motor4.run(BACKWARD);
  delay(1000);
  motor1.run(FORWARD);
  motor2.run(FORWARD);
  motor3.run(FORWARD);
  motor4.run(FORWARD);
}

void turnLeft() {
  motor1.run(BACKWARD);
  motor2.run(BACKWARD);
  motor3.run(FORWARD);
  motor4.run(FORWARD);
  delay(1000);
  motor1.run(FORWARD);
  motor2.run(FORWARD);
  motor3.run(FORWARD);
  motor4.run(FORWARD);
}
```

Successive commands and methods implemented as above which are self explanatory .

The logic behind the rotation of car can be explained by concept of torque.



Schematic Diagram for taking left . Wheels on right side move in forward while wheels on left side move backward for taking left turn.