Ultrasonic obstacle-avoidance car

First, welding lead on the four ports of the DC geared motor. Then, put the different parts of the car together. After that, fix the Dc motor driven plate on the holes of the inside car.

The next step is to wiring. The right-front wheel's lead should be like this: The negative pole is above while the positive pole is below. The negative pole should be connected with OUT1 port while the positive pole should be connected with OUT2 port; The rear-right wheel's lead should be like this: The negative pole is above while the positive pole is below. The negative pole should be connected with OUT2 while the positive pole should be connected with the OUT1 port.

The left-front wheel's lead should be like this: The positive pole is above while the negative pole is below. The negative pole should be connected with OUT4 port while the positive pole should be connected with OUT3 port. The left-rear wheel's lead should be like this: The positive pole is above while the negative pole is below. The negative pole should be connected with OUT3 port while the positive pole should be connected with OUT4 port.

After that, put the sensor V5 board into the MEGA2560. Connect IN1, IN2, IN3, IN4 to the 8,9,6,7 of the pin respectively. Afterward, link the Power supply interface 5V and GND of the chip of the motor driven plate to any V or G pin of the sensor V5 board. Connect the power supply interface VCC and GND of the motor driven plate with 9-16V battery or Dc power supply. Fix the Ultrasonic module and the MEGA2560 onto the car. Link Ultrasonic's VCC and GND port to any V or G pin of the sensor V5. Trig and Echo port should be connected with 3, 2 port of the sensor V5 board respectively.

The last step is to download the texting program to the MEGA2560 so that you can see the car automatically move forward and avoid the obstacle.

```
const int EchoPin = 2; // Ultrasonic signal input
const int TrigPin = 3; // Ultrasonic signal output

const int leftmotorpin1 = 8; //signal output of DC geared motor
const int leftmotorpin2 = 9;
const int rightmotorpin1 = 6;
const int rightmotorpin2 = 7;

int currDist = 0;        // distance

void setup() {
   pinMode(EchoPin, INPUT);
   for (int pinindex = 3; pinindex < 8; pinindex++) {
      pinMode(pinindex, OUTPUT); // set pins 3 to 10 as outputs
   }
}

void loop() {
   currDist = MeasuringDistance(); //measure front distance

   if(currDist >20) {
      nodanger();
```

```
    }
    else if(currDist <10){
        backup();
        randTrun();
    }
    else {
        //whichway();
        randTrun();
    }
}

//measure distance, unit:cm
  long MeasuringDistance() {
    long duration;
    //pinMode(TrigPin, OUTPUT);
    digitalWrite(TrigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(TrigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(TrigPin, LOW);

    //pinMode(EchoPin, INPUT);
    duration = pulseIn(EchoPin, HIGH);

    return duration / 29 / 2;
}

// forward
void nodanger() {
    digitalWrite(leftmotorpin1, HIGH);
    digitalWrite(leftmotorpin2, LOW);
    digitalWrite(rightmotorpin1, HIGH);
    digitalWrite(rightmotorpin2, LOW);
    delay(1000);
    return;
}

//backward
  void backup() {
    digitalWrite(leftmotorpin1, LOW);
    digitalWrite(leftmotorpin2, HIGH);
    digitalWrite(rightmotorpin1, LOW);
    digitalWrite(rightmotorpin2, HIGH);
    delay(500);
```

```
}

void totalhalt() {
    digitalWrite(leftmotorpin1, HIGH);
    digitalWrite(leftmotorpin2, HIGH);
    digitalWrite(rightmotorpin1, HIGH);
    digitalWrite(rightmotorpin2, HIGH);
    return;
    delay(500);
}

//turn left
void body_lturn() {
    digitalWrite(leftmotorpin1, LOW);
    digitalWrite(leftmotorpin2, HIGH);
    digitalWrite(rightmotorpin1, HIGH);
    digitalWrite(rightmotorpin2, LOW);
    delay(500);
    totalhalt();
}

//turn right
  void body_rturn() {
    digitalWrite(leftmotorpin1, HIGH);
    digitalWrite(leftmotorpin2, LOW);
    digitalWrite(rightmotorpin1, LOW);
    digitalWrite(rightmotorpin2, HIGH);
    delay(500);
    totalhalt();
}

void randTrun(){
    long randNumber;
    randomSeed(analogRead(0));
    randNumber = random(0, 10);
    if(randNumber > 5) {
        body_rturn();
    }
    else
    {
        body_lturn();
    }
}
```