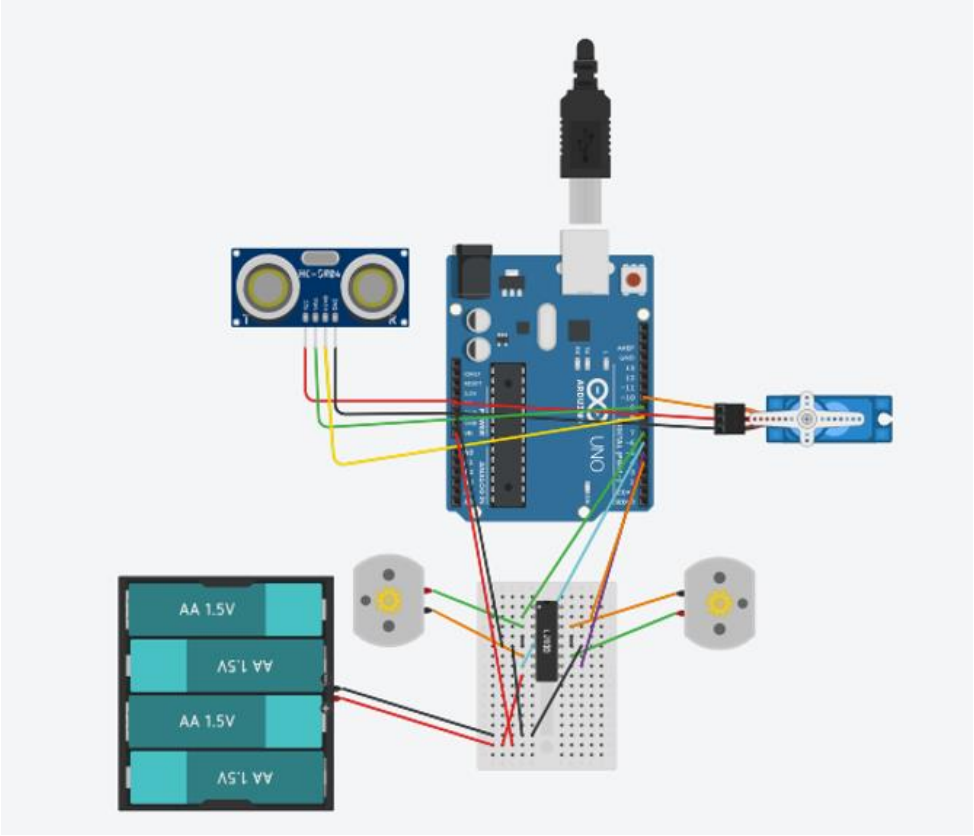
1. Component List

	A	B	C	D	E	F	G
1	OBSTACLE AVOIDING BOT						
2	COMPNENET LIST						
3							
4	NAME OF COMPONENT	VALUE /RANGE	QUANTITY	Link	Cost	Total C	
5	Rover Chasis	-	1	https://robu.in/product/cha	Rs 126.00	126	
6	Wheels	-	4	https://robu.in/product/rob	Rs 28.00	112	
7	Caster Wheel		1	https://robu.in/product/wh	Rs 42.00	42	
8	Bo Motors	6V, 20RPM	2	https://robu.in/product/20r	Rs 96.00	192	
9	Servo Motor	SG90	1	https://robu.in/product/tov	Rs 174.00	174	
10	Arduino	UNO	1	https://robu.in/product/ard	Rs 583.68	583	
11	4AA Battery Holder	-	1	https://robu.in/product/4-x	Rs 38	38	
12	Li-Ion Battery Orange A grade	ISR 18650	4	https://robu.in/product/ora	Rs 299	1196	
13	Ultrasonic Sensor	HC-SR04	1	https://robu.in/product/hc-	Rs 76.00	76	
14	Jumper Wires	M-F	40	https://robu.in/product/ma	Rs 39.00	39	
15	Motor Driver	L298N	1	https://robu.in/product/l29	Rs 143.00	143	
16					Total Amt	2721	
17							

2. Circuit Diagram



***used L239D instead of L239 Motor driver board (tinkercad limitation)**

3. Code

obstacle_bot_code.ino

```ino
#include <Servo.h>
Servo Myservo;
#define trigPin 9           // Trig Pin Of HC-SR04
#define echoPin 8        // Echo Pin Of HC-SR04
#define MLa 4                 //left motor 1st pin
#define MLb 5                //left motor 2nd pin
#define MRa 6              //right motor 1st pin
#define MRb 7              //right motor 2nd pin
long duration, distance;

void setup() {
  Serial.begin(9600);
  pinMode(MLa, OUTPUT);     // Set Motor Pins As O/P
  pinMode(MLb, OUTPUT);
  pinMode(MRa, OUTPUT);
  pinMode(MRb, OUTPUT);
  pinMode(trigPin, OUTPUT);       // Set Trig Pin As O/P To Transmit Waves
  pinMode(echoPin, INPUT);        //Set Echo Pin As I/P To Receive Reflected Waves
  Myservo.attach(10);
}
void loop()
{
  Serial.begin(9600);
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);       // Transmit Waves For 10us
  delayMicroseconds(10);
  duration = pulseIn(echoPin, HIGH);        // Receive Reflected Waves
  distance = duration / 58.2;                    // Get Distance
  Serial.println(distance);
  delay(10);
  if (distance > 15)              // Condition For Absence Of Obstacle
  {
    Myservo.write(90);
    digitalWrite(MRb, HIGH);       // Move Forward
    digitalWrite(MRa, LOW);
    digitalWrite(MLb, HIGH);
    digitalWrite(MLa, LOW);
  }
  else if ((distance < 10)&&(distance > 0))           // Condition For Presence Of Obstacle
  {
    digitalWrite(MRb, LOW);     //Stop
    digitalWrite(MRa, LOW);
    digitalWrite(MLb, LOW);
    digitalWrite(MLa, LOW);
    delay(100);

    Myservo.write(0);
    delay(500);
    Myservo.write(180);
    delay(500);
    Myservo.write(90);
    delay(500);

    digitalWrite(MRb, LOW);     // Move Backward
    digitalWrite(MRa, HIGH);
    digitalWrite(MLb, LOW);
    digitalWrite(MLa, HIGH);
    delay(500);
    digitalWrite(MRb, LOW);         //Stop
    digitalWrite(MRa, LOW);
    digitalWrite(MLb, LOW);
    digitalWrite(MLa, LOW);
    delay(100);
    digitalWrite(MRb, HIGH);      // Move Left
    digitalWrite(MRa, LOW);
    digitalWrite(MLa, LOW);
    digitalWrite(MLb, LOW);
    delay(500);
  }

}
```