Practical 7
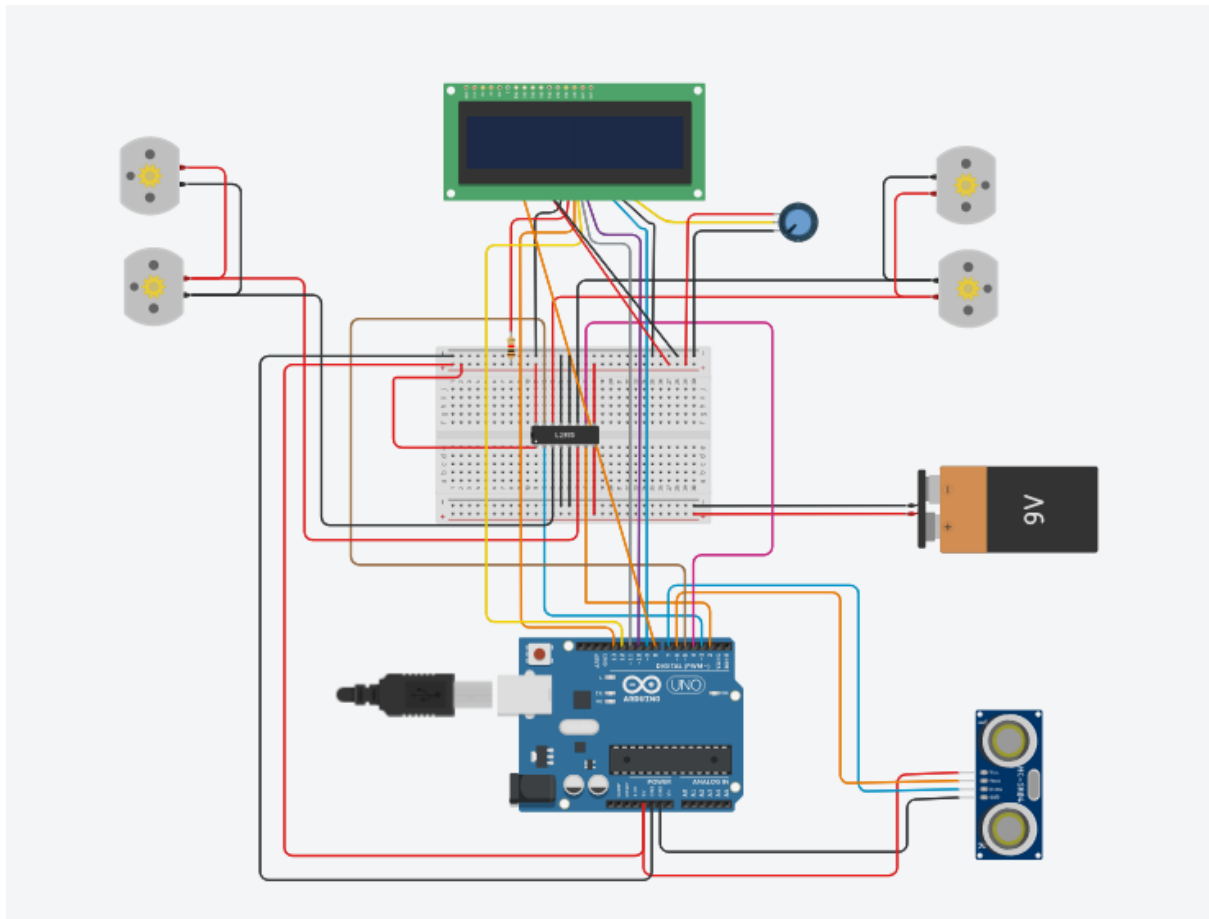# Aim: - Create an obstacle avoidance behavior for robot and test it.

Components:
1) 4 Dc Motors
2) 9 Volt Battery
3) Arduino UNO R3
4) Breadboard Small
5) 1 Resistor
6) 1 Ultrasonic Sensor
7) 1 H-bridge Motor Driver
8) 1 Potentiometer

Step 1: Connect all the device with the wires and complete the circuit (Using TinkerCad)

Step 2: Now we have to write the code for the following circuit

Code:

```cpp
#include <LiquidCrystal.h>

// Initialize the library with the numbers of the interface pins
LiquidCrystal lcd(8, 9, 10, 11, 12, 13);

long cm, duration;
const int echoPin = 7;
const int trigPin = 6;
const int lm1 = 2;
const int lm2 = 3;
const int rm1 = 4;   // Corrected pin name from rm3 to rm1
const int rm2 = 5;   // Corrected pin name from rm4 to rm2

void setup() {
  pinMode(lm1, OUTPUT);
  pinMode(lm2, OUTPUT);
  pinMode(rm1, OUTPUT);
  pinMode(rm2, OUTPUT);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  Serial.begin(9600);
  lcd.begin(16, 2);
}

void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(5);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  // Converting time into distance in centimetres
  cm = duration * 0.034 / 2;

  if (cm < 20) {
    stop_bot();
    delay(2000);
    go_back();
    delay(2000);
```

```
    stop_again();
    delay(1000);
    go_left();
    delay(1000);
  } else {
    go_straight();
    delay(1000);
  }

  Serial.print("Distance: CM ");
  Serial.println(cm);
}

void go_straight() {
  lcd.setCursor(0, 0);
  lcd.print("NOTHING AHEAD");
  lcd.setCursor(0, 1);
  lcd.print("MOVING FORWARD");
    digitalWrite(lm1, HIGH);
    digitalWrite(lm2, LOW);
    digitalWrite(rm1, HIGH);
    digitalWrite(rm2, LOW);
}

void go_back() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("TAKING REVERSE");
  lcd.setCursor(0, 1);
  lcd.print(cm);
  digitalWrite(lm1, LOW);
  digitalWrite(lm2, HIGH);
  digitalWrite(rm1, LOW);
  digitalWrite(rm2, HIGH);
}

void stop_bot() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("SOMETHING AHEAD");
  lcd.setCursor(0, 1);
  lcd.print("STOP!");
```

```
    digitalWrite(lm1, LOW);
    digitalWrite(lm2, LOW);
    digitalWrite(rm1, LOW);
    digitalWrite(rm2, LOW);
}

void stop_again() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("BREAK FOR TURN");
  digitalWrite(lm1, LOW);
  digitalWrite(lm2, LOW);
  digitalWrite(rm1, LOW);
  digitalWrite(rm2, LOW);
}

void go_left() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("TURNING LEFT");
  lcd.setCursor(0, 1);
  lcd.print(cm);
  digitalWrite(lm1, LOW);
  digitalWrite(lm2, LOW);
  digitalWrite(rm1, HIGH);
  digitalWrite(rm2, LOW);
}

void go_right() {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("TURNING RIGHT");
  lcd.setCursor(0, 1);
  lcd.print(cm);
  digitalWrite(lm1, HIGH);
  digitalWrite(lm2, LOW);
  digitalWrite(rm1, LOW);
  digitalWrite(rm2, LOW);
}
```
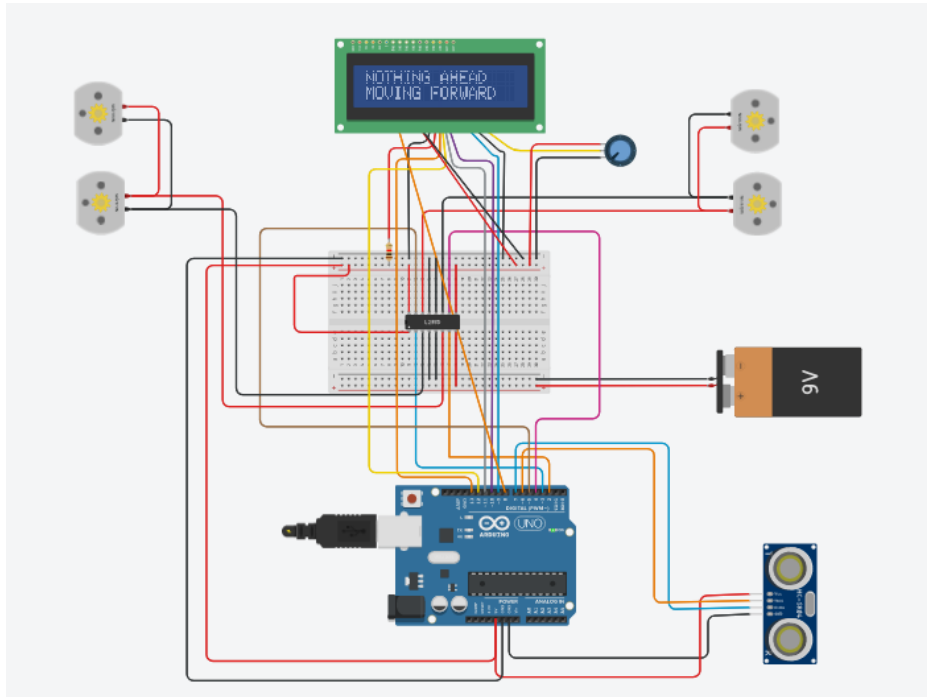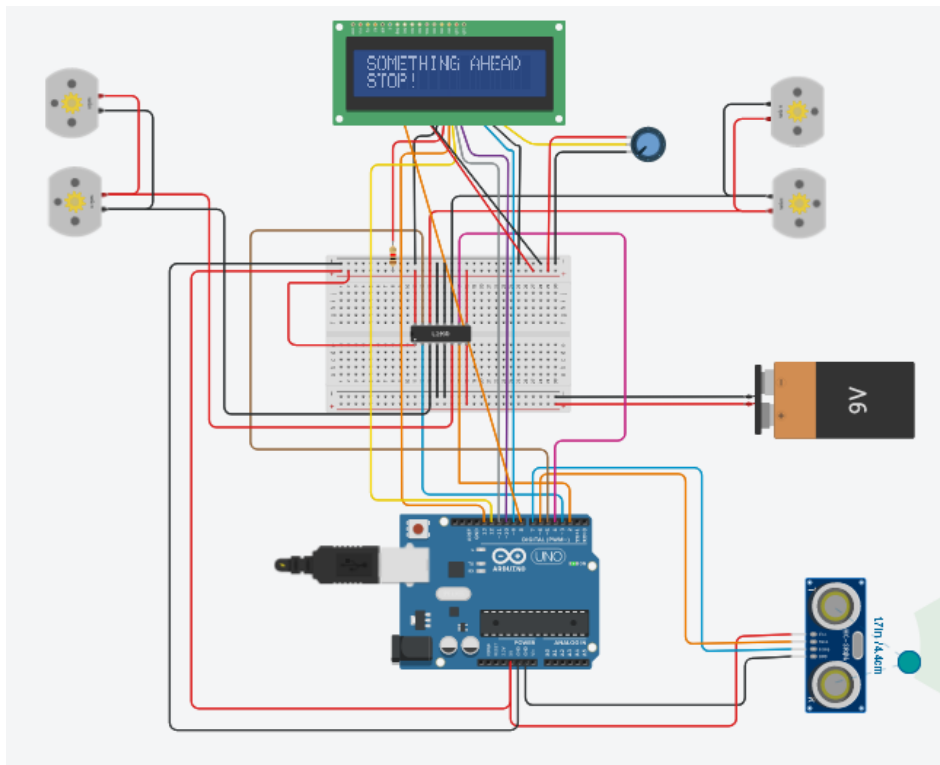
Step 3: Once after writing the code, click on the Start Simulation Button and now you can see the circuit is running

Robot Moving Ahead as there is no obstacle

Robot stops as the sensor detects objects in front.

Now the Robot is going in reverse direction