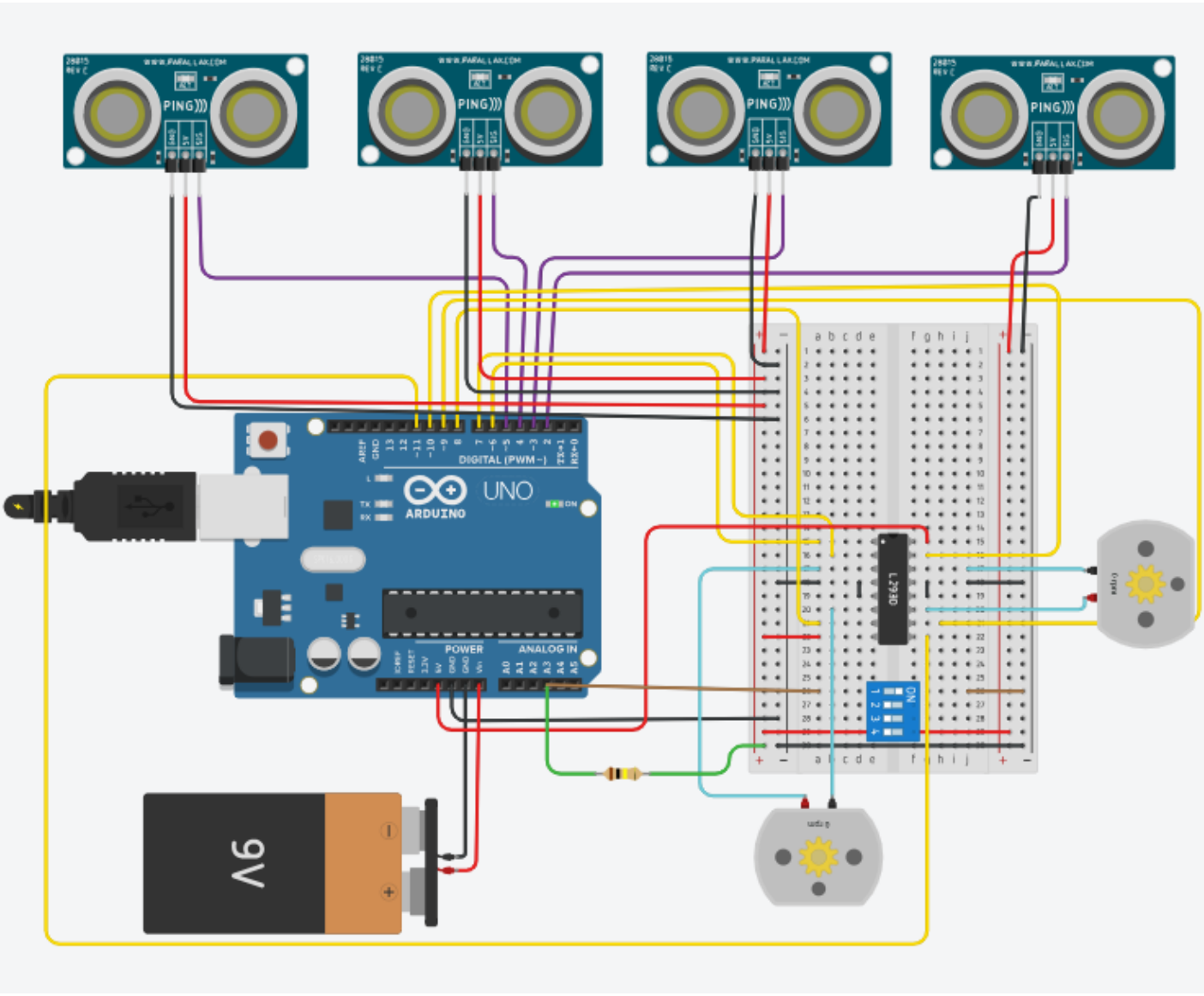
Aram al-shahateet



```
#define trigPins {2, 4, 6, 8}
#define echoPins {3, 5, 7, 9}

#define ENA 10
#define IN1 11
#define IN2 12
#define IN3 A0
#define IN4 A1
#define ENB A2

#define SWITCH A3

long duration;
int distances[4];
int trig[] = trigPins;
int echo[] = echoPins;

void setup() {
  for (int i = 0; i < 4; i++) {
    pinMode(trig[i], OUTPUT);
    pinMode(echo[i], INPUT);
  }

  pinMode(ENA, OUTPUT);
  pinMode(ENB, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);

  pinMode(SWITCH, INPUT_PULLUP);
```

```
  Serial.begin(9600);
}

int getDistance(int trig, int echo) {
  digitalWrite(trig, LOW);
  delayMicroseconds(2);
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  duration = pulseIn(echo, HIGH, 30000);
  return duration > 0 ? (duration * 0.034 / 2) : 1000;
}

void readSensors() {
  for (int i = 0; i < 4; i++) {
    distances[i] = getDistance(trig[i], echo[i]);
    delay(50);
  }
}

void move(int speedA, int speedB, bool in1, bool in2, bool in3, bool in4) {
  analogWrite(ENA, speedA);
  analogWrite(ENB, speedB);
  digitalWrite(IN1, in1);
  digitalWrite(IN2, in2);
  digitalWrite(IN3, in3);
  digitalWrite(IN4, in4);
}

void moveForward() { move(200, 200, HIGH, LOW, HIGH, LOW); }
void moveBackward() { move(200, 200, LOW, HIGH, LOW, HIGH); }
void turnRight() { move(200, 200, HIGH, LOW, LOW, HIGH); }
```

```
void turnLeft() { move(200, 200, LOW, HIGH, HIGH, LOW); }
void stopRobot() { move(0, 0, LOW, LOW, LOW, LOW); }

void loop() {
  if (digitalRead(SWITCH) == LOW) {
    stopRobot();
    return;
  }

  readSensors();

  Serial.print("Front: "); Serial.print(distances[0]);
  Serial.print(" Right: "); Serial.print(distances[1]);
  Serial.print(" Left: "); Serial.print(distances[2]);
  Serial.print(" Back: "); Serial.println(distances[3]);

  if (distances[0] > 40) {
    moveForward();
  } else if (distances[1] > 40) {
    turnRight();
    delay(map(distances[1], 0, 40, 500, 100)); // Dynamic delay
  } else if (distances[2] > 40) {
    turnLeft();
    delay(map(distances[2], 0, 40, 500, 100));
  } else if (distances[3] > 40) {
    moveBackward();
    delay(map(distances[3], 0, 40, 500, 100));
  } else {
    stopRobot();
  }

  delay(100);
}
```