INDEX

PRACTICAL NO. 1

Introduction to Arduino

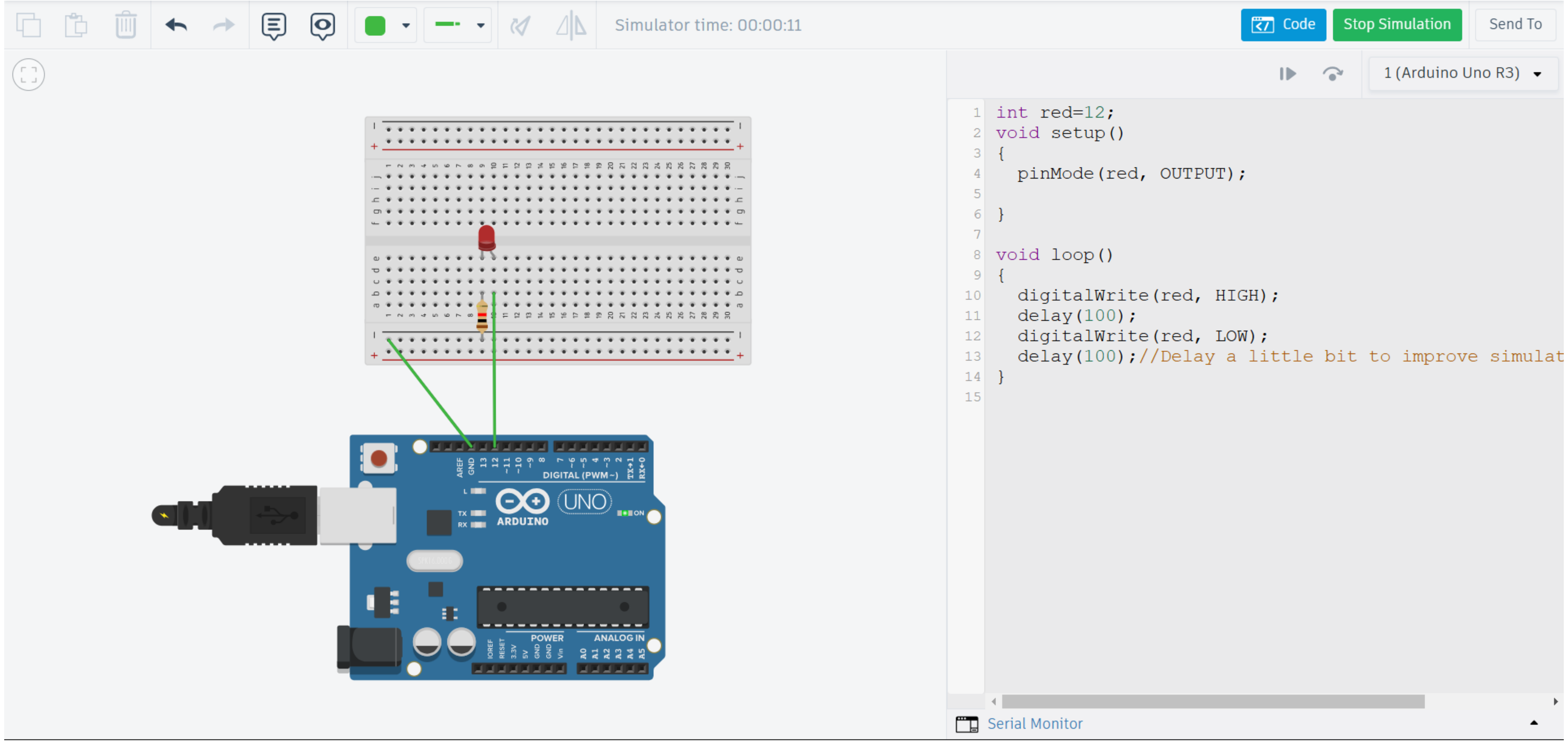
Introduction to Arduino circuits and breadboarding

Blinking of LEDs

Code:

```
int red=12;
void setup()
{
  pinMode(red, OUTPUT);
}
void loop()
{
  digitalWrite(red, HIGH);
  delay(1000);
  digitalWrite(red, LOW);
  delay(1000);//Delay a little bit to improve simulation performance
}
```

Output:



```
int red=12;
void setup()
{
  pinMode(red, OUTPUT);

}

void loop()
{
  digitalWrite(red, HIGH);
  delay(100);
  digitalWrite(red, LOW);
  delay(100);//Delay a little bit to improve simulat
}
```

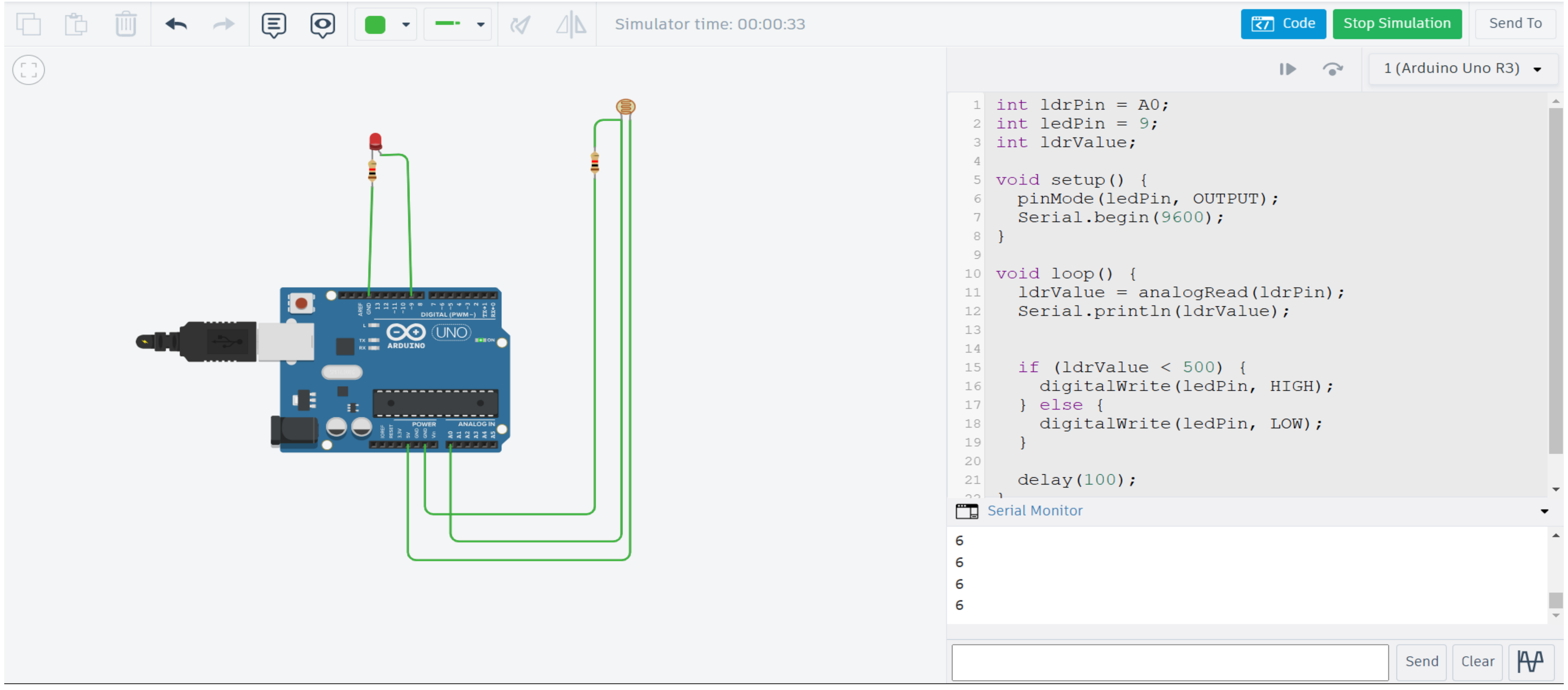
PRACTICAL NO. 2

Program using Light Sensitive Sensors

Code:

```
int ldrPin = A0;
int ledPin = 9;
int ldrValue;
void setup() {
 pinMode(ledPin, OUTPUT);
 Serial.begin(9600);
}
void loop() {
 ldrValue = analogRead(ldrPin);
 Serial.println(ldrValue);
 if (ldrValue < 500) {
  digitalWrite(ledPin, HIGH);
 } else {
  digitalWrite(ledPin, LOW);
 }
 delay(100);
}
```

Output:



```
int ldrPin = A0;
int ledPin = 9;
int ldrValue;

void setup() {
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  ldrValue = analogRead(ldrPin);
  Serial.println(ldrValue);


  if (ldrValue < 500) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }

  delay(100);
}
```

```
6
6
6
6
```

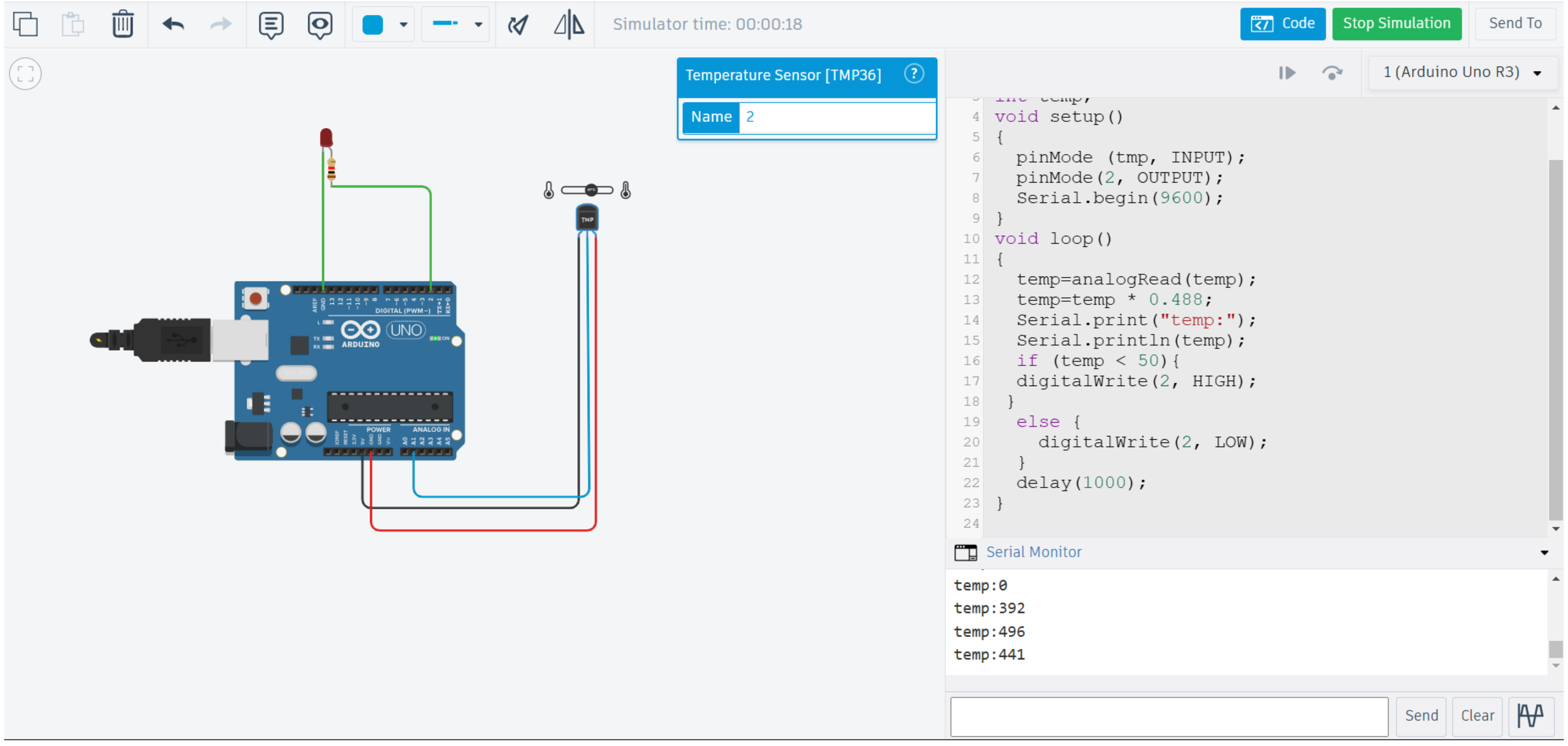
PRACTICAL NO. 3

Program using temperature sensors

Code:

```
byte tmp = A1;
byte led= 2;
int temp;
void setup()
{
 pinMode (tmp, INPUT);
 pinMode(2, OUTPUT);
 Serial.begin(9600);
}
void loop()
{
 temp=analogRead(temp);
 temp=temp * 0.488;
 Serial.print("temp:");
 Serial.println(temp);
 if (temp < 50){
 digitalWrite(2, HIGH);
}
 else {
  digitalWrite(2, LOW);
 }
 delay(1000);
}
```

Output:



```
int temp;
void setup()
{
  pinMode (tmp, INPUT);
  pinMode(2, OUTPUT);
  Serial.begin(9600);
}
void loop()
{
  temp=analogRead(temp);
  temp=temp * 0.488;
  Serial.print("temp:");
  Serial.println(temp);
  if (temp < 50){
  digitalWrite(2, HIGH);
 }
  else {
    digitalWrite(2, LOW);
  }
  delay(1000);
}
```

```
temp:0
temp:392
temp:496
temp:441
```

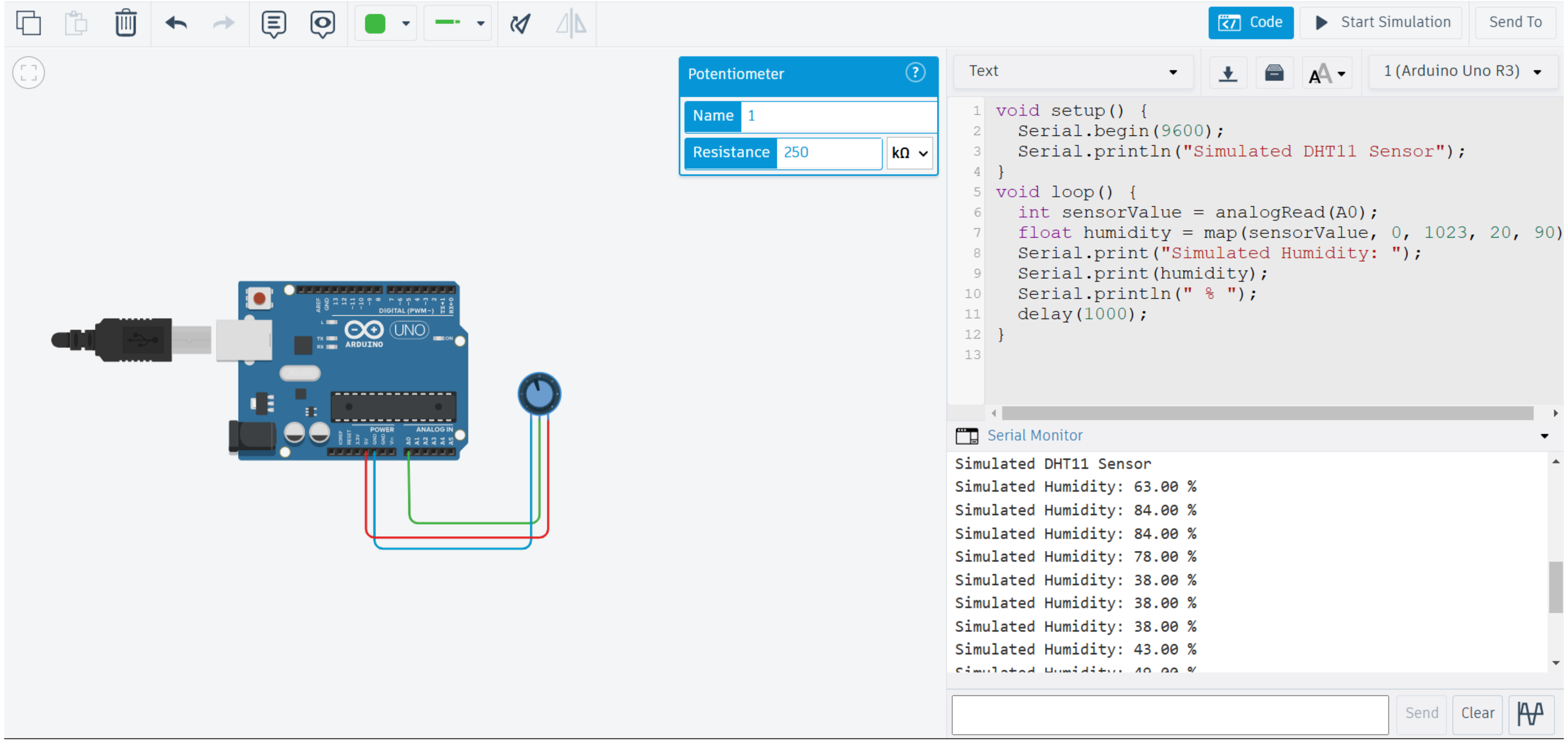
PRACTICAL NO. 4

Program using humidity sensors

Code:

```
void setup() {
  Serial.begin(9600);
  Serial.println("Simulated DHT11 Sensor");
}
void loop() {
  int sensorValue = analogRead(A0);
  float humidity = map(sensorValue, 0, 1023, 20, 90);
  Serial.print("Simulated Humidity: ");
  Serial.print(humidity);
  Serial.println(" % ");
  delay(1000);
}
```

Output:



Potentiometer

Name: 1

Resistance: 250 kΩ

```
void setup() {
  Serial.begin(9600);
  Serial.println("Simulated DHT11 Sensor");
}
void loop() {
  int sensorValue = analogRead(A0);
  float humidity = map(sensorValue, 0, 1023, 20, 90)
  Serial.print("Simulated Humidity: ");
  Serial.print(humidity);
  Serial.println(" % ");
  delay(1000);
}
```

```
Simulated DHT11 Sensor
Simulated Humidity: 63.00 %
Simulated Humidity: 84.00 %
Simulated Humidity: 84.00 %
Simulated Humidity: 78.00 %
Simulated Humidity: 38.00 %
Simulated Humidity: 38.00 %
Simulated Humidity: 38.00 %
Simulated Humidity: 43.00 %
Simulated Humidity: 49.00 %
```

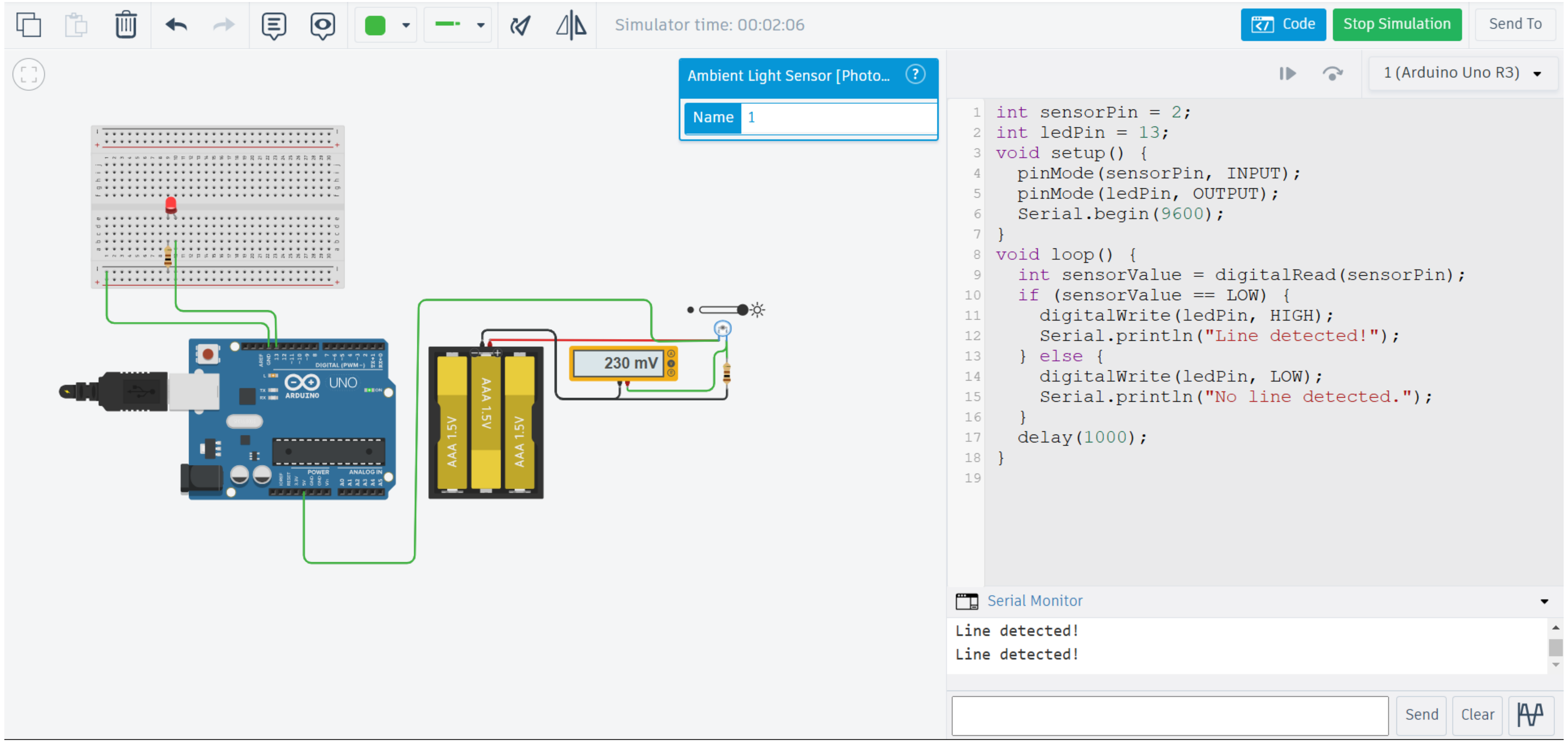
PRACTICAL NO. 5

Programs using Line tracking sensors

Code:

```
int sensorPin = 2;
int ledPin = 13;
void setup() {
  pinMode(sensorPin, INPUT);
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}
void loop() {
  int sensorValue = digitalRead(sensorPin);
  if (sensorValue == LOW) {
    digitalWrite(ledPin, HIGH);
    Serial.println("Line detected!");
  } else {
    digitalWrite(ledPin, LOW);
    Serial.println("No line detected.");
  }
  delay(1000);
}
```

Output:



Name 1

```
int sensorPin = 2;
int ledPin = 13;
void setup() {
  pinMode(sensorPin, INPUT);
  pinMode(ledPin, OUTPUT);
  Serial.begin(9600);
}
void loop() {
  int sensorValue = digitalRead(sensorPin);
  if (sensorValue == LOW) {
    digitalWrite(ledPin, HIGH);
    Serial.println("Line detected!");
  } else {
    digitalWrite(ledPin, LOW);
    Serial.println("No line detected.");
  }
  delay(1000);
}
```

```
Line detected!
Line detected!
```

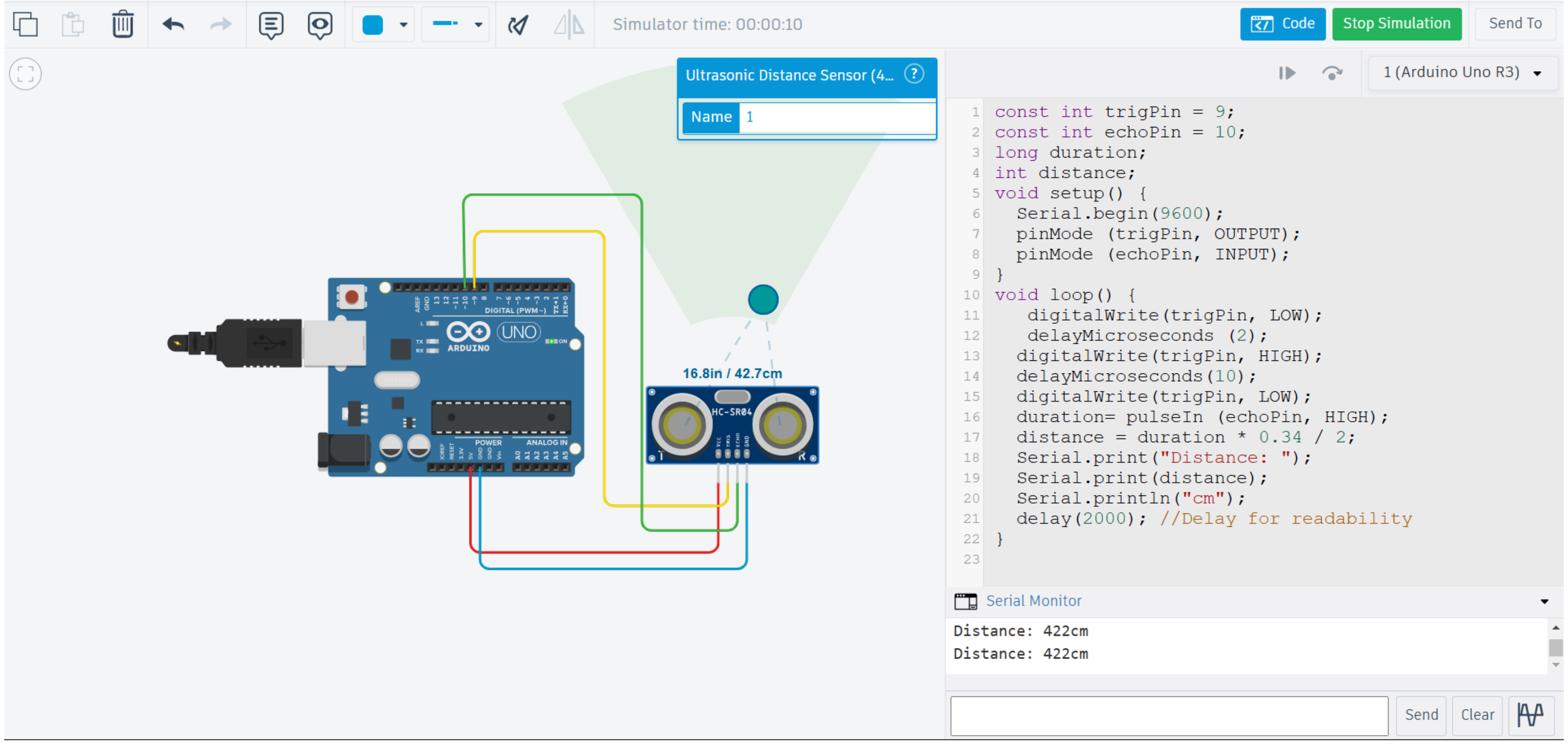
PRACTICAL NO. 6

Programs using Ultrasonic Sensors

Code:

```
const int trigPin = 9;
const int echoPin = 10;
long duration;
int distance;
void setup() {
  Serial.begin(9600);
  pinMode (trigPin, OUTPUT);
  pinMode (echoPin, INPUT);
}
void loop() {
   digitalWrite(trigPin, LOW);
   delayMicroseconds (2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration= pulseIn (echoPin, HIGH);
  distance = duration * 0.34 / 2;
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println("cm");
  delay(2000); //Delay for readability
}
```

Output:



```
const int trigPin = 9;
const int echoPin = 10;
long duration;
int distance;
void setup() {
  Serial.begin(9600);
  pinMode (trigPin, OUTPUT);
  pinMode (echoPin, INPUT);
}
void loop() {
   digitalWrite(trigPin, LOW);
   delayMicroseconds (2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration= pulseIn (echoPin, HIGH);
  distance = duration * 0.34 / 2;
  Serial.print("Distance: ");
  Serial.print(distance);
  Serial.println("cm");
  delay(2000); //Delay for readability
}
```

```
Distance: 422cm
Distance: 422cm
```

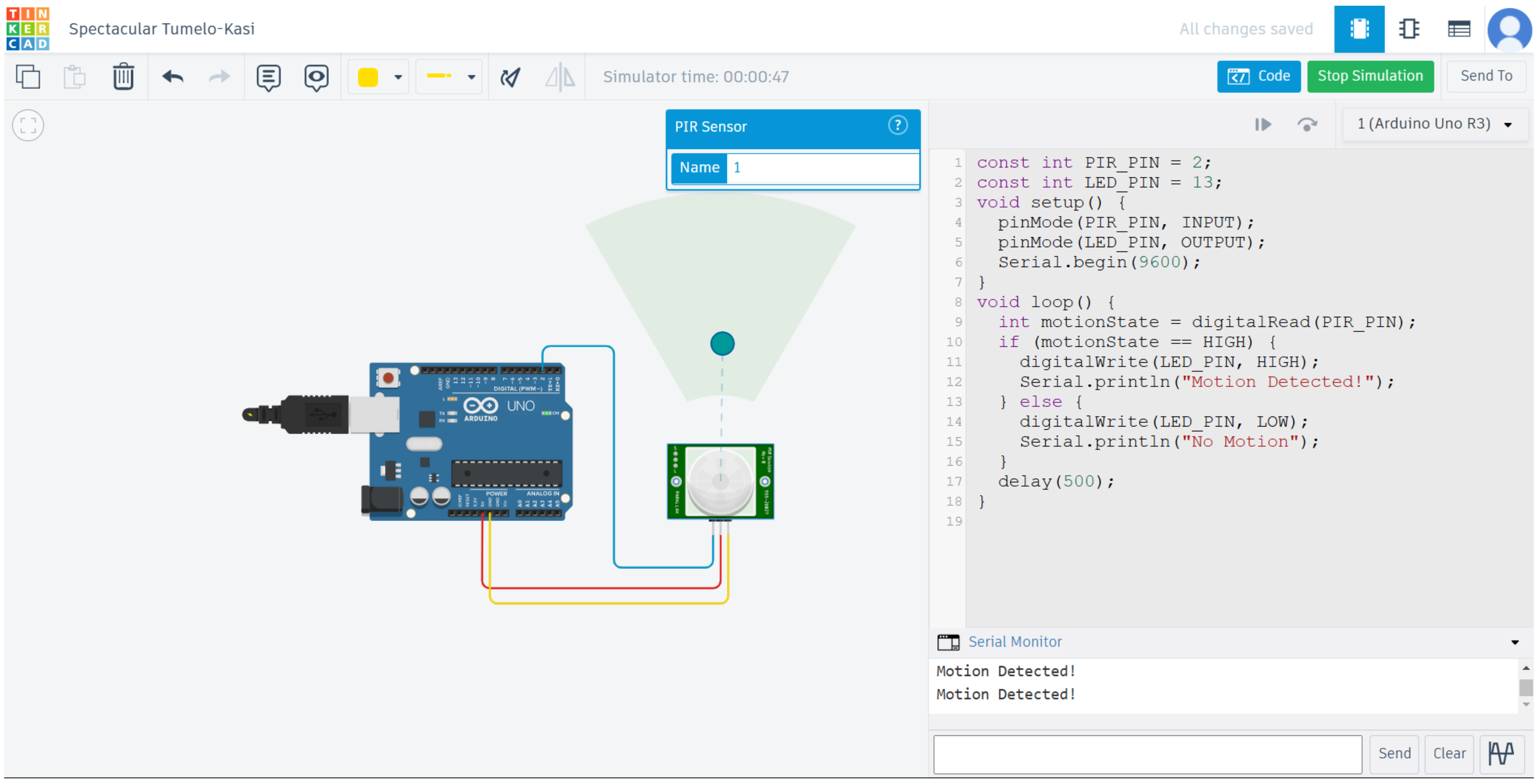
PRACTICAL NO. 7

Programs using digital infrared motion sensors

Code:

```
const int PIR_PIN = 2;
const int LED_PIN = 13;
void setup() {
 pinMode(PIR_PIN, INPUT);
 pinMode(LED_PIN, OUTPUT);
 Serial.begin(9600);
}
void loop() {
 int motionState = digitalRead(PIR_PIN);
 if (motionState == HIGH) {
   digitalWrite(LED_PIN, HIGH);
   Serial.println("Motion Detected!");
 } else {
   digitalWrite(LED_PIN, LOW);
   Serial.println("No Motion");
 }
 delay(500);
}
```

Output:



```
const int PIR_PIN = 2;
const int LED_PIN = 13;
void setup() {
  pinMode(PIR_PIN, INPUT);
  pinMode(LED_PIN, OUTPUT);
  Serial.begin(9600);
}
void loop() {
  int motionState = digitalRead(PIR_PIN);
  if (motionState == HIGH) {
    digitalWrite(LED_PIN, HIGH);
    Serial.println("Motion Detected!");
  } else {
    digitalWrite(LED_PIN, LOW);
    Serial.println("No Motion");
  }
  delay(500);
}
```

```
Motion Detected!
Motion Detected!
```

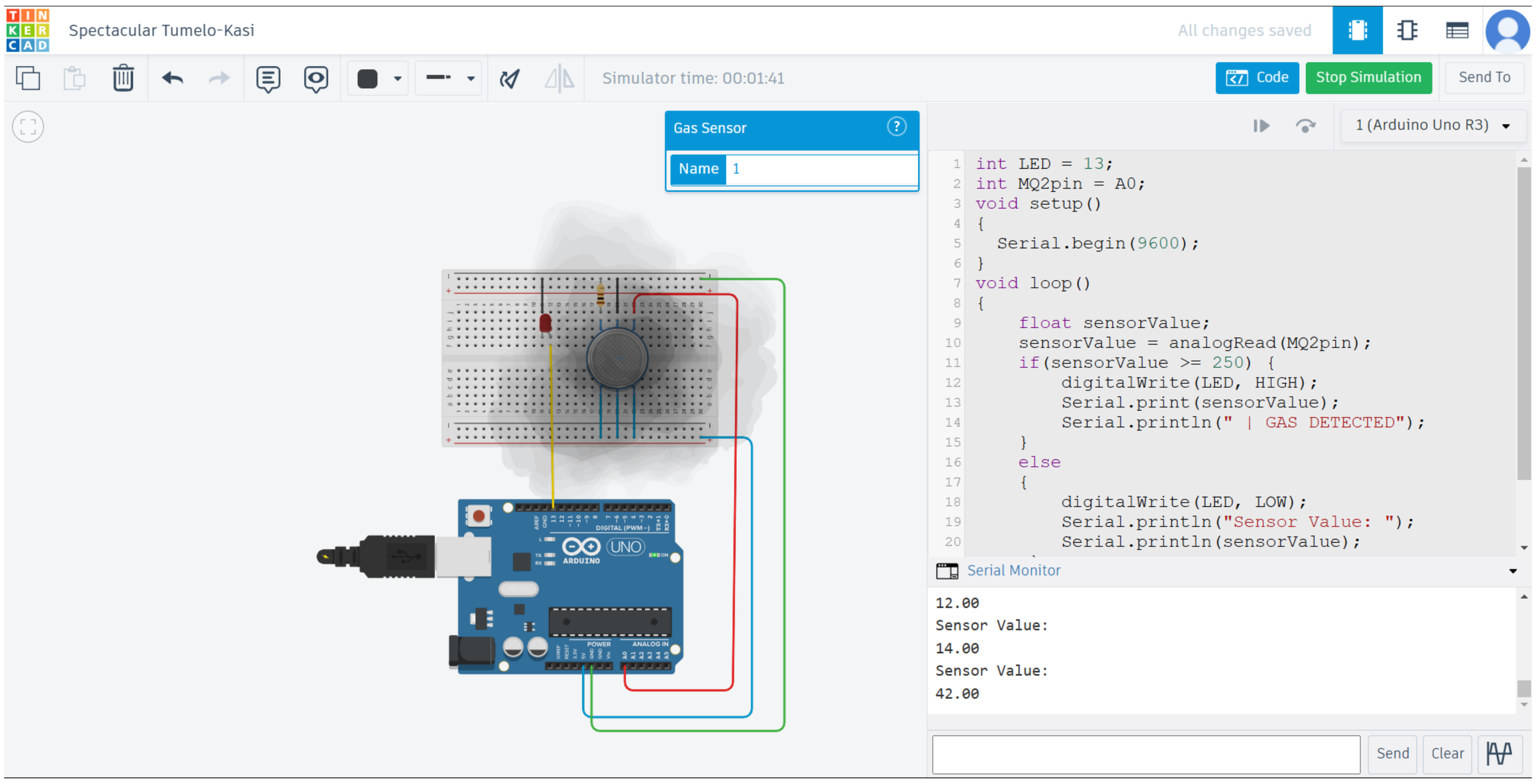
PRACTICAL NO. 8

Programs using gas sensors

Code:

```
int LED = 13;
int MQ2pin = A0;
void setup()
{
 Serial.begin(9600);
}
void loop()
{
   float sensorValue;
   sensorValue = analogRead(MQ2pin);
   if(sensorValue >= 250)
   {
      digitalWrite(LED, HIGH);
      Serial.print(sensorValue);
      Serial.println(" | GAS DETECTED");
   }
   else
   {
      digitalWrite(LED, LOW);
      Serial.println("Sensor Value: ");
      Serial.println(sensorValue);
    }
    delay(1000);
}
```

Output:



```
int LED = 13;
int MQ2pin = A0;
void setup()
{
  Serial.begin(9600);
}
void loop()
{
    float sensorValue;
    sensorValue = analogRead(MQ2pin);
    if(sensorValue >= 250) {
        digitalWrite(LED, HIGH);
        Serial.print(sensorValue);
        Serial.println(" | GAS DETECTED");
    }
    else
    {
        digitalWrite(LED, LOW);
        Serial.println("Sensor Value: ");
        Serial.println(sensorValue);
```

```
12.00
Sensor Value:
14.00
Sensor Value:
42.00
```

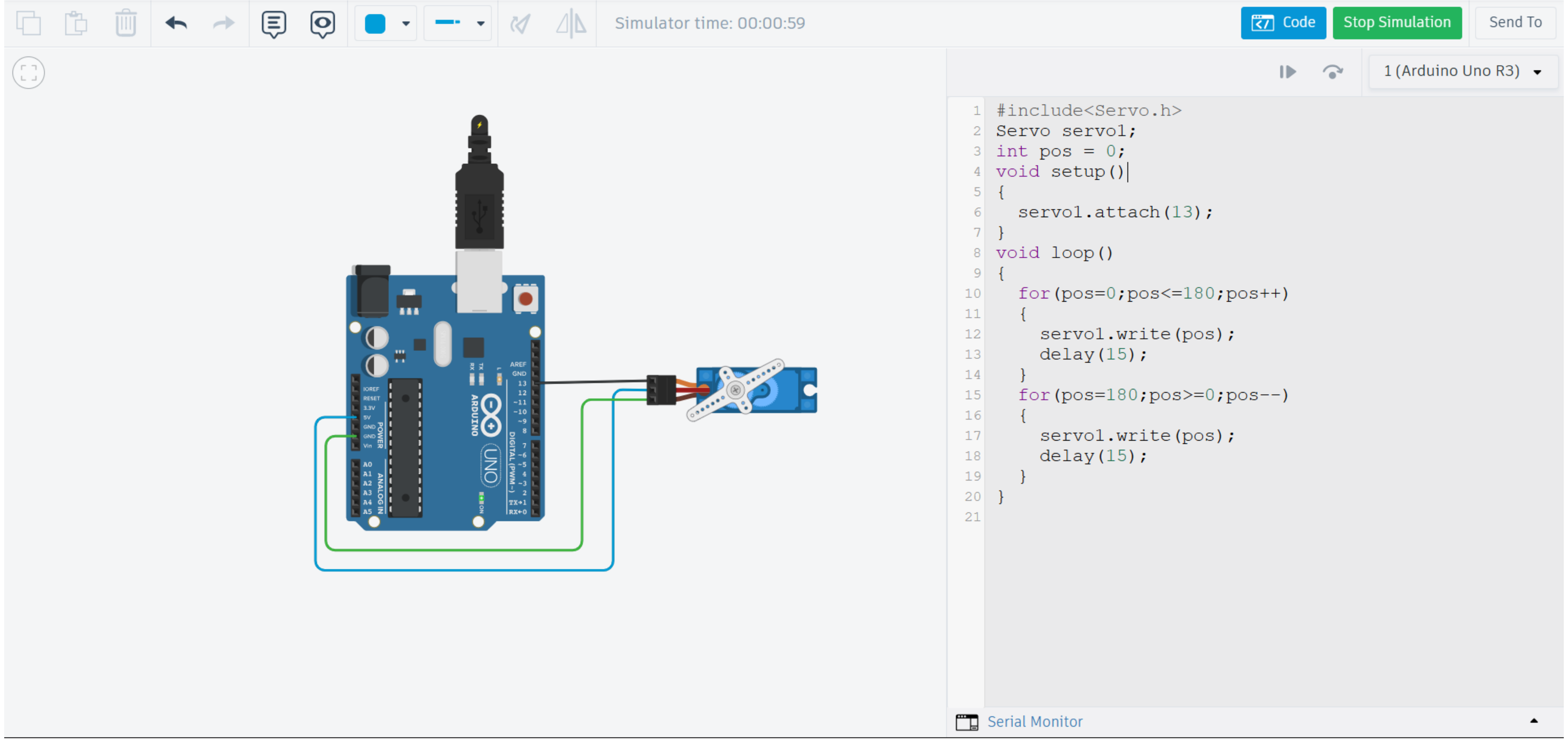
PRACTICAL NO. 9

Programs using servo motors

Code:

```
#include<Servo.h>
Servo servo1;
int pos = 0;
void setup()
{
 servo1.attach(13);
}
void loop()
{
 for(pos=0;pos<=180;pos++)
 {
  servo1.write(pos);
  delay(15);
 }
 for(pos=180;pos>=0;pos--)
 {
  servo1.write(pos);
  delay(15);
 }
}
```

Output:



```
#include<Servo.h>
Servo servo1;
int pos = 0;
void setup()
{
  servo1.attach(13);
}
void loop()
{
  for(pos=0;pos<=180;pos++)
  {
    servo1.write(pos);
    delay(15);
  }
  for(pos=180;pos>=0;pos--)
  {
    servo1.write(pos);
    delay(15);
  }
}
```

PRACTICAL NO. 10

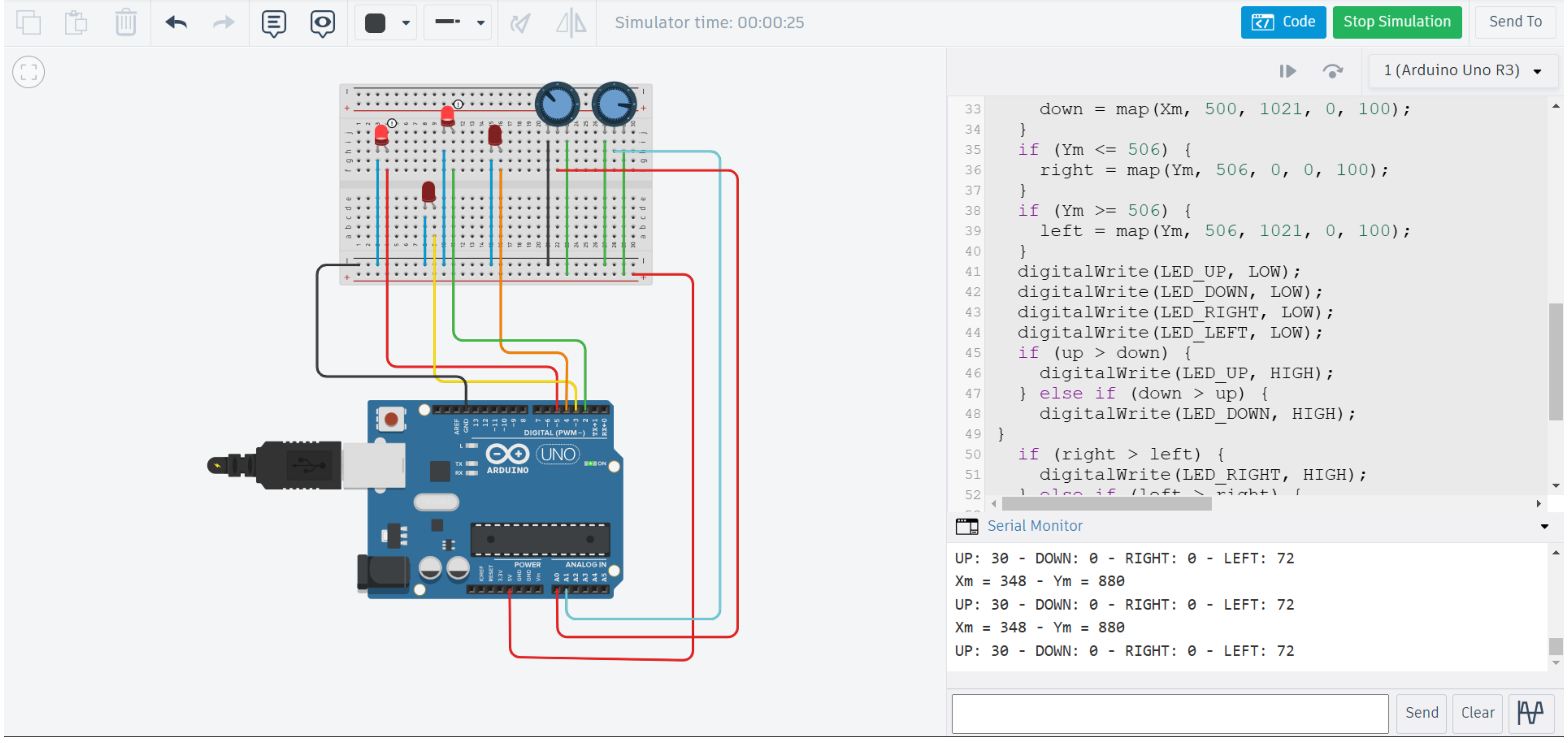
Programs making Joystick with Arduino

Code:

```
int x[50] = {0};
int y[50] = {0};
int LED_UP = 2;
int LED_DOWN = 3;
int LED_RIGHT = 4;
int LED_LEFT = 5;
void setup() {
  Serial.begin(9600);
  pinMode(LED_UP, OUTPUT);
  pinMode(LED_DOWN, OUTPUT);
  pinMode(LED_RIGHT, OUTPUT);
  pinMode(LED_LEFT, OUTPUT);
}
void loop() {
  long int sumX = 0;
  long int sumY = 0;
  for (int i = 0; i < 50; i++) {
    x[i] = analogRead(A0);
    sumX += x[i];
    y[i] = analogRead(A1);
    sumY += y[i];
  }
  int Xm = sumX / 50;
  int Ym = sumY / 50;
  int up = 0;
  int down = 0;
  int right = 0;
  int left = 0;
  if (Xm <= 500) {
    up = map(Xm, 500, 0, 0, 100);
  }
  if (Xm >= 500) {
    down = map(Xm, 500, 1021, 0, 100);
  }
  if (Ym <= 506) {
    right = map(Ym, 506, 0, 0, 100);
  }
  if (Ym >= 506) {
    left = map(Ym, 506, 1021, 0, 100);
  }
  digitalWrite(LED_UP, LOW);
```

```
 digitalWrite(LED_DOWN, LOW);
 digitalWrite(LED_RIGHT, LOW);
 digitalWrite(LED_LEFT, LOW);
 if (up > down) {
  digitalWrite(LED_UP, HIGH);
 } else if (down > up) {
  digitalWrite(LED_DOWN, HIGH);
}
 if (right > left) {
  digitalWrite(LED_RIGHT, HIGH);
 } else if (left > right) {
  digitalWrite(LED_LEFT, HIGH);
 }
 String phrase1 = "Xm = " + String(Xm) + " - Ym = " + String(Ym);
 String phrase2 = "UP: " + String(up) + " - DOWN: " + String(down) + " - RIGHT: " + String(right) +
" - LEFT: " + String(left);
 Serial.println(phrase1);
 Serial.println(phrase2);
 delay(1000);
}
```

Output:



```
      down = map(Xm, 500, 1021, 0, 100);
    }
    if (Ym <= 506) {
      right = map(Ym, 506, 0, 0, 100);
    }
    if (Ym >= 506) {
      left = map(Ym, 506, 1021, 0, 100);
    }
    digitalWrite(LED_UP, LOW);
    digitalWrite(LED_DOWN, LOW);
    digitalWrite(LED_RIGHT, LOW);
    digitalWrite(LED_LEFT, LOW);
    if (up > down) {
      digitalWrite(LED_UP, HIGH);
    } else if (down > up) {
      digitalWrite(LED_DOWN, HIGH);
  }
    if (right > left) {
      digitalWrite(LED_RIGHT, HIGH);
    } else if (left > right) {
```

```
UP: 30 - DOWN: 0 - RIGHT: 0 - LEFT: 72
Xm = 348 - Ym = 880
UP: 30 - DOWN: 0 - RIGHT: 0 - LEFT: 72
Xm = 348 - Ym = 880
UP: 30 - DOWN: 0 - RIGHT: 0 - LEFT: 72
```