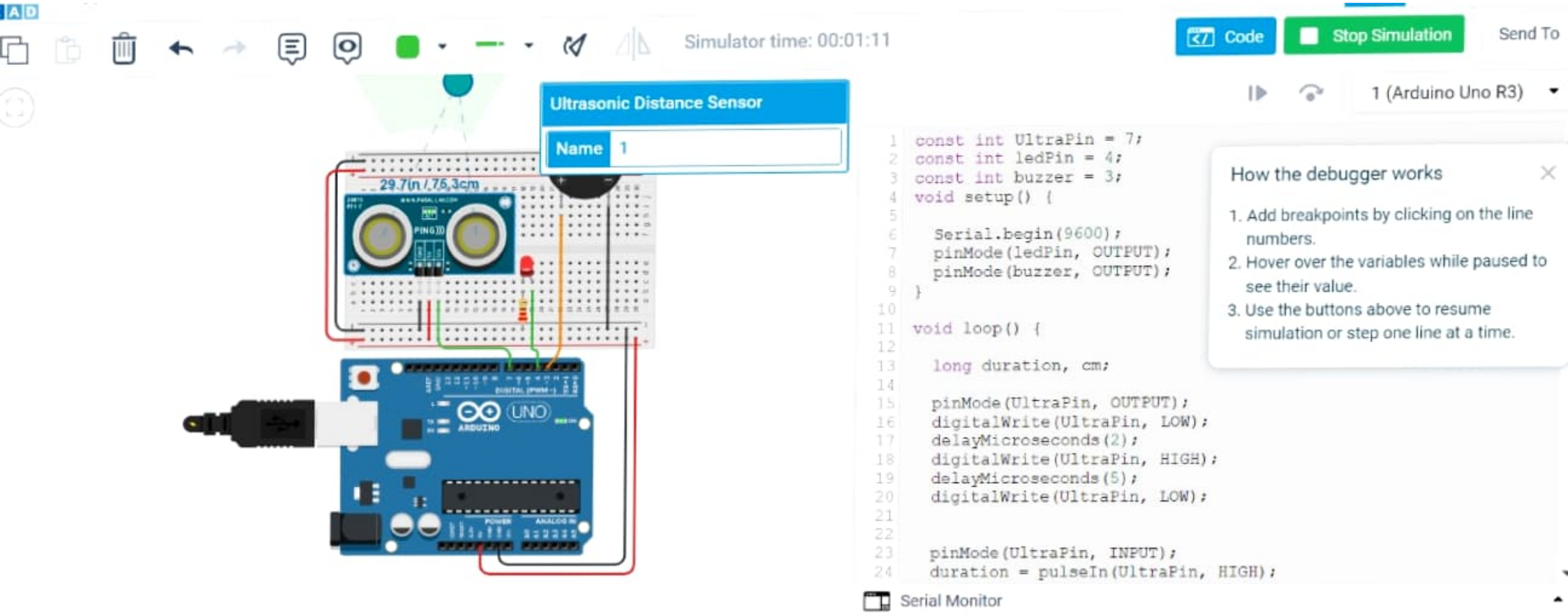
Simulator time: 00:01:11
Code
Stop Simulation
Send To
1 (Arduino Uno R3)
Ultrasonic Distance Sensor
Name 1
29.7in / 75.3cm
const int UltraPin = 7;
const int ledPin = 4;
const int buzzer = 3;
void setup() {
Serial.begin(9600);
pinMode(ledPin, OUTPUT);
pinMode(buzzer, OUTPUT);
}
void loop() {
long duration, cm;
pinMode(UltraPin, OUTPUT);
digitalWrite(UltraPin, LOW);
delayMicroseconds(2);
digitalWrite(UltraPin, HIGH);
delayMicroseconds(5);
digitalWrite(UltraPin, LOW);
pinMode(UltraPin, INPUT);
duration = pulseIn(UltraPin, HIGH);
How the debugger works
1. Add breakpoints by clicking on the line numbers.
2. Hover over the variables while paused to see their value.
3. Use the buttons above to resume simulation or step one line at a time.
Serial Monitor

```
Const int UltraPin = 7;
Const int ledPin = 4;
Const int buzzer = 3;
Void setup() {

  Serial.begin(9600);
  pinMode(ledPin, OUTPUT);
  pinMode(buzzer, OUTPUT);
}

Void loop() {

  Long duration, cm;

  pinMode(UltraPin, OUTPUT);
  digitalWrite(UltraPin, LOW);
  delayMicroseconds(2);
  digitalWrite(UltraPin, HIGH);
  delayMicroseconds(5);
  digitalWrite(UltraPin, LOW);


  pinMode(UltraPin, INPUT);
  duration = pulseIn(UltraPin, HIGH);

  // convert the time into a distance
  Cm = microsecondsToCentimeters(duration);

  // Print the distance
```

```
  Serial.print("Distance: ");
  Serial.print(cm);
  Serial.print("cm");
  Serial.println();

  // Turn on the LED if the object is too close:
  If(cm < 100) {
    digitalWrite(ledPin, HIGH);
    digitalWrite(buzzer, HIGH);
  }
  Else {
    digitalWrite(ledPin, LOW);
    digitalWrite(buzzer, LOW);
  }

  Delay(100);
}


Long microsecondsToCentimeters(long microseconds) {

  Return microseconds / 29 / 2;
}
```