

05/28/24  
Wesley  
Coda.

## Lidar for Object Detection

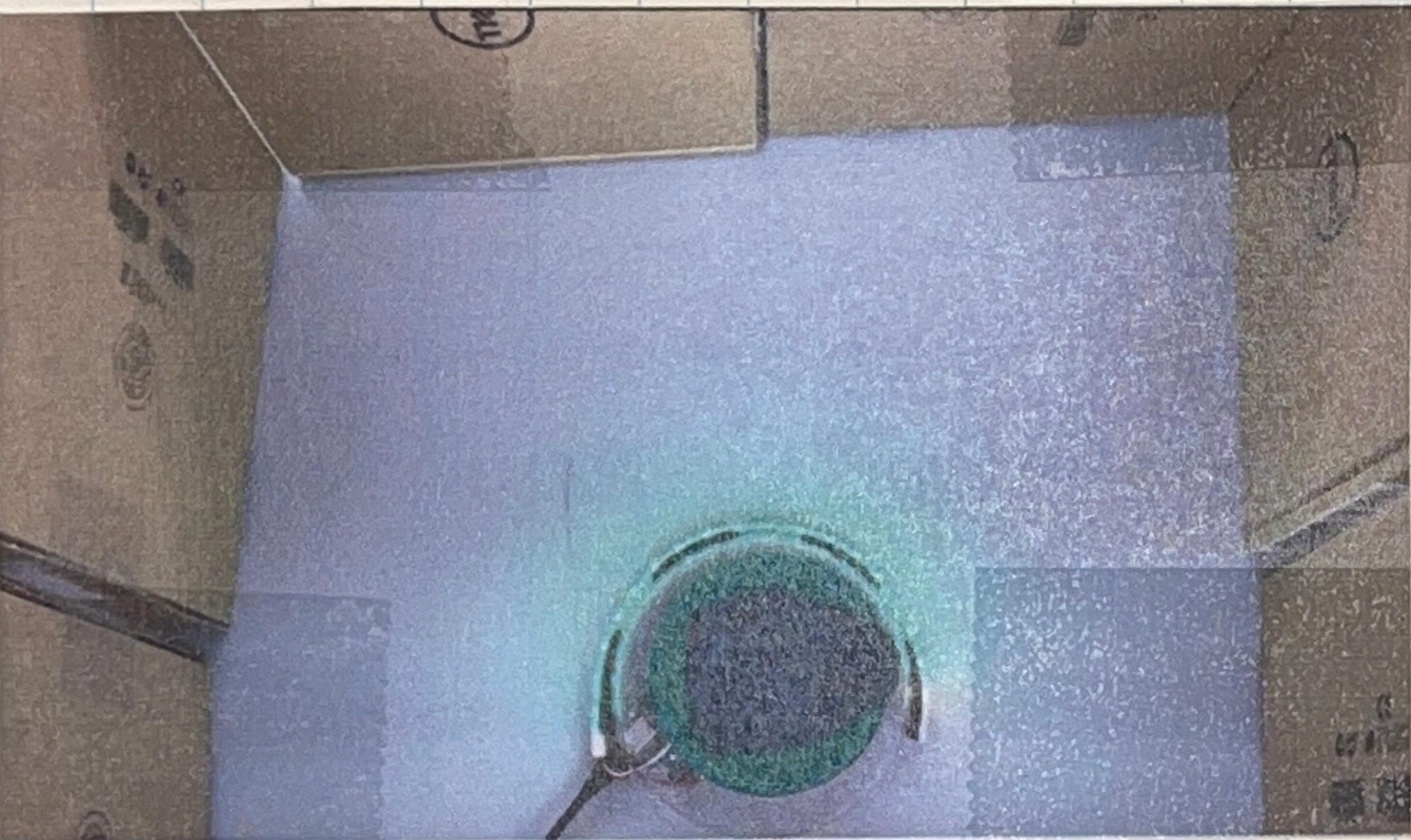
25

Install the robo peak arduino library;  
[https://github.com/robopeak/rplidar\\_arduino](https://github.com/robopeak/rplidar_arduino)

Sketch → include Library → Add .ZIP

### Objective

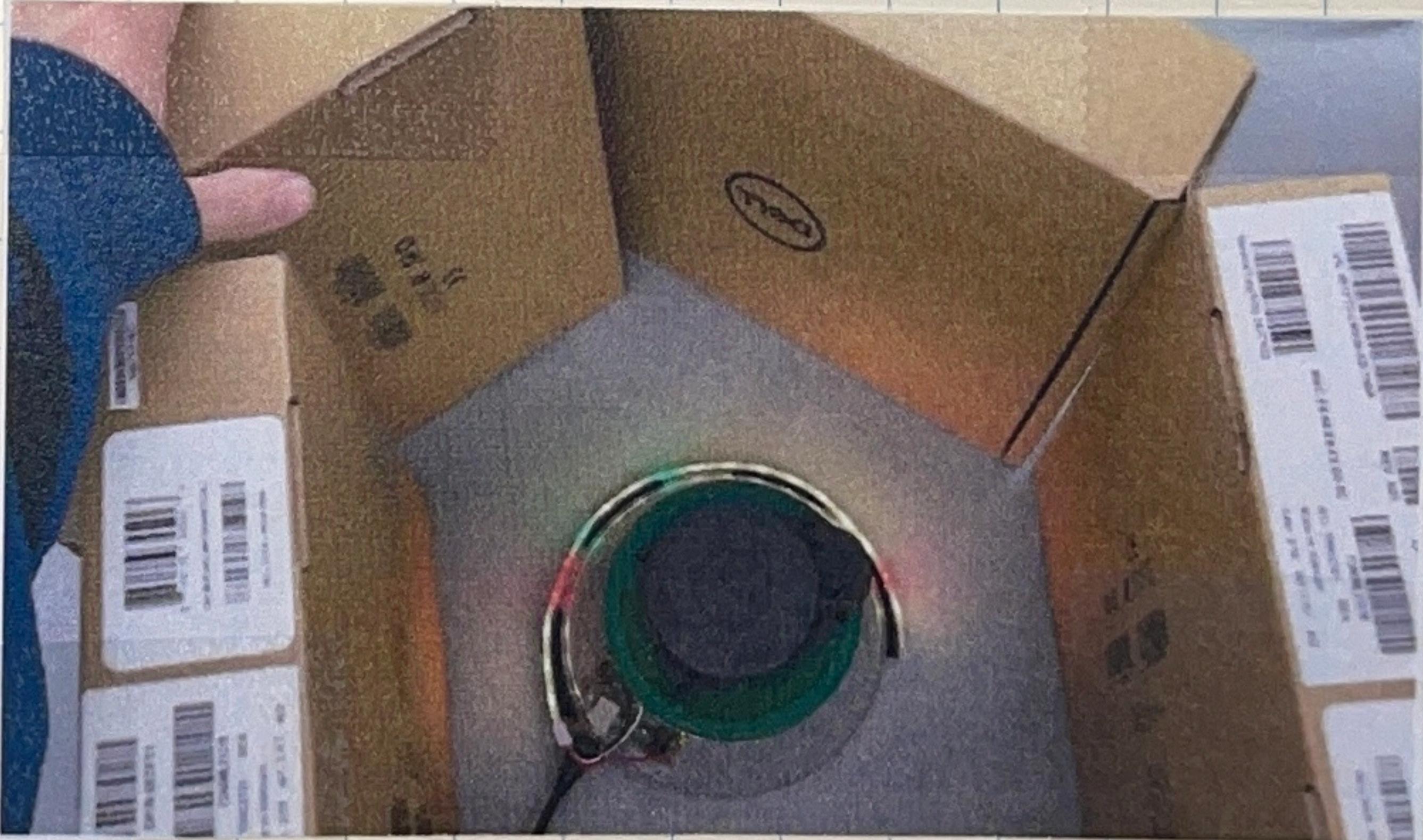
Detect objects that are close or far away  
and light up an LED that relates to the  
distance.



### Results

~~Results~~ Results for  
objects far away

Results for objects  
at a medium distance



Results for objects  
very close.

Code

```

1 #include <RPLidar.h>
2 #include <Adafruit_NeoPixel.h>
3
4 #define LED_PIN 8
5 #define LED_COUNT 19 // These variables are for the neo pixels.
6
7 Adafruit_NeoPixel strip(LED_COUNT, LED_PIN, NEO_GRB+NEO_KHZ800);
8
9 RPLidar lidar;
10 #define RPLIDAR_MOTOR 3 // PWM Pin for motor control.
11
12 // Let's keep an array that has a distance for every integer angle.
13 float distance[360]; // Can access indexes 0-359.
14 // Let's look at 10 rotations before we average our data.
15 int numRotations = 0;
16 int targetRotations = 10;

18 void setup()
19 {
20   Serial.begin(115200);
21   lidar.begin(Serial);
22   pinMode(RPLIDAR_MOTOR, OUTPUT);
23
24   strip.begin();
25   strip.setBrightness(50);
26 }
27 void printData()
28 {
29   // This method shows each angle and their corresponding distance
30   for (int i=0; i<=360; i++)
31   {
32     Serial.print(i);
33     Serial.print(" ");
34     Serial.println(distance[i]);
35   }
36 }

103 void loop()
104 {
105   if (IS_OK(lidar.waitPoint()))
106   {
107     RPLidarMeasurement currentMeasure = lidar.getCurrentPoint();
108     if (currentMeasure.startBit == 1)
109     {
110       numRotations++; ↑ looking for 10 rotations.
111       if (numRotations >= targetRotations)
112       {
113         printData();
114         numRotations = 0;
115         lightLED(); *→ all the cool stuff is in here
116       }
117     }
118   else
119   {
120     distance[int(currentMeasure.angle)] = currentMeasure.distance;
121   }
122 }

123 else
124 {
125   analogWrite(RPLIDAR_MOTOR, 0); // Stop the motor
126   rplidar_response_device_info_t info;
127   Serial.println("---");
128   Serial.println(IS_OK(lidar.waitPoint()));
129   Serial.println(IS_OK(lidar.getDeviceInfo(info, 100)));
130   if (IS_OK(lidar.getDeviceInfo(info, 100)))
131   {
132     lidar.startScan();
133     analogWrite(RPLIDAR_MOTOR, 255);
134     delay(1000);
135   }
136 }
137 }
```

} handle an  
error state  
and restart.

05/28/24  
W. values  
object

# Lidar for Object Detection

27

```
38 void lightLED()
39 {
40
41 // We know we need 19 averages. -> i= (0-18) 10 degrees for each pixel.
42 for(int i=0; i<19; i++)
43 {
44 // We keep a sum and how many value we added to our average.
45 float sum = 0;
46 float averageNum = 0;
47 float averageDistance = 0.00;
48
49 // Here, we need to look at the proper subset of our data.
50 // For iteration one, we want to look at indexes 0-9 (degrees)
51 // For iteration two, we want to look at indexes 10-19 (degrees)
52
53 // So we start by multiplying which average we are on by 10. -> i*10
54 // we go up to an index of i*10 + 9. So that we get 10 total values.
```

} inner for loop explanation.

```
55
56 for(int a = i*10; a<=i*10+9; a++) *→ look at proper subset.
57 {
58 // We are ignoring 0 values since we don't know
59 // this 0 is from being too close, or that the lidar
60 // didn't get data there.
61 if (distance[a] != 0.00)
62 {
63 sum += distance[a];
64 averageNum++;
65 }
66 }
67 // If we actually added a non zero value to our sum.
68 if (averageNum != 0)
69 {
70 // Get the average distance
71 averageDistance = sum/averageNum;
72 }
73 else
74 {
75 // Otherwise, set it to 0 to say either
76 // 1. We didn't actually get distances at those angles.
77 // 2. Something is way too close to the sensor.
78 averageDistance = 0.00;
79 }
80
81 // Set the strip color based on the average distance.
82 Serial.println(averageDistance);
83 if (averageDistance == 0.00) // White for 0's
84 {
85 strip.setPixelColor(i, strip.Color(255,255,255));
86 }
87 else if (averageDistance < 500) // Red for too close!
88 {
89 strip.setPixelColor(i, strip.Color(255,0,0));
90 }
91 else if (averageDistance < 1000) // Yellow for getting close
92 {
93 strip.setPixelColor(i, strip.Color(255,165,0));
94 }
95 else // Green for not close enough
96 {
97 strip.setPixelColor(i, strip.Color(0,255,0));
98 }
99 }
100 strip.show();
101 }
```

} Light up the LED's