```
1  /*
2    Ping))) Sensor
3
4    This sketch reads a PING))) ultrasonic rangefinder and returns the distance
5    to the closest object in range. To do this, it sends a pulse to the sensor  ⇁
       to
6    initiate a reading, then listens for a pulse to return. The length of the
7    returning pulse is proportional to the distance of the object from the       ⇁
       sensor.
8
9    The circuit:
10     - +V connection of the PING))) attached to +5V
11     - GND connection of the PING))) attached to ground
12     - SIG connection of the PING))) attached to digital pin 7
13
14   created 3 Nov 2008
15   by David A. Mellis
16   modified 30 Aug 2011
17   by Tom Igoe
18
19   This example code is in the public domain.
20
21   http://www.arduino.cc/en/Tutorial/Ping
22
23   Modifiziert Lutz Bär 24.10.2020
24   http://dl1rlb.net
25
26   Abstandsensor in Verbindung mit einem 8*8 LED Matrix CJMCU.
27   In Coroanazeiten soll er bei unterschreiten des Abstandes
28   ein Smiley zeichnen
29
30 */
31
32 //8x8Matrix
33 #include <Adafruit_NeoPixel.h>
34 #define PIN 9 // DIN Datenpin
35 Adafruit_NeoPixel strip = Adafruit_NeoPixel(64, PIN, NEO_RGB + NEO_KHZ800);//
36
37 //Abstandssensor
38 // this constant won't change. It's the pin number of the sensor's output:
39
40 const int pingPin = 7; // Pin vom Sensor
41
42 const int Piep = 4; // PIN vom Piepser
43
44 const int Abstand = 50; // Abstand in cm
45
46
47 void setup()
48 {
49     // initialize serial communication:
50     Serial.begin(9600);
51     pinMode(Piep, OUTPUT);
52     //=========8x8Matrix=========
53     strip.begin();
54     strip.show();
```

```csharp
55       //======8x8Matrix=========
56  }
57
58  //==========8x8Matrix=========
59  void smiley(uint32_t color)
60  {
61      for (byte i = 0; i < 9; i = i + 1)
62      {
63          //Auge links 4 LEDs
64          strip.setPixelColor(uint8_t(9), color);
65          strip.setPixelColor(uint8_t(10), color);
66          strip.setPixelColor(uint8_t(17), color);
67          strip.setPixelColor(uint8_t(18), color);
68          //Auge rechts 4 LEDs
69          strip.setPixelColor(uint8_t(13), color);
70          strip.setPixelColor(uint8_t(14), color);
71          strip.setPixelColor(uint8_t(21), color);
72          strip.setPixelColor(uint8_t(22), color);
73
74          // LED an für Mund lachen
75          strip.setPixelColor(uint8_t(32), color);
76          strip.setPixelColor(uint8_t(41), color);
77          strip.setPixelColor(uint8_t(50), color);
78          strip.setPixelColor(uint8_t(51), color);
79          strip.setPixelColor(uint8_t(52), color);
80          strip.setPixelColor(uint8_t(53), color);
81          strip.setPixelColor(uint8_t(46), color);
82          strip.setPixelColor(uint8_t(39), color);
83
84          // LEDs ausschalten vom Mund traurig, sonst würden die LEDs
                weiterleuchten
85          strip.setPixelColor(uint8_t(48), strip.Color(0, 0, 0));
86          strip.setPixelColor(uint8_t(34), strip.Color(0, 0, 0));
87          strip.setPixelColor(uint8_t(27), strip.Color(0, 0, 0));
88          strip.setPixelColor(uint8_t(28), strip.Color(0, 0, 0));
89          strip.setPixelColor(uint8_t(37), strip.Color(0, 0, 0));
90          strip.setPixelColor(uint8_t(35), strip.Color(0, 0, 0));
91          strip.setPixelColor(uint8_t(36), strip.Color(0, 0, 0));
92          strip.setPixelColor(uint8_t(55), strip.Color(0, 0, 0));
93          strip.show();
94          delay(15);
95      }
96  }
97
98  void bad(uint32_t color)
99  {
100     for (byte i = 0; i < 8; i = i + 1)
101     {
102         //Auge links 3 LEDs
103         strip.setPixelColor(uint8_t(9), color);
104         strip.setPixelColor(uint8_t(10), color);
105         strip.setPixelColor(uint8_t(17), color);
106
107         //Auge rechts 3 LEDs
108         strip.setPixelColor(uint8_t(13), color);
109         strip.setPixelColor(uint8_t(14), color);
```

```
110          strip.setPixelColor(uint8_t(22), color);
111
112          // LED einschalten für Traurig
113          strip.setPixelColor(uint8_t(48), color);
114          strip.setPixelColor(uint8_t(41), color);
115          strip.setPixelColor(uint8_t(34), color);
116          strip.setPixelColor(uint8_t(35), color);
117          strip.setPixelColor(uint8_t(36), color);
118          strip.setPixelColor(uint8_t(37), color);
119          strip.setPixelColor(uint8_t(46), color);
120          strip.setPixelColor(uint8_t(55), color);
121
122          // LED ausschalten von lachen, sonst würden die LEDs weiterleuchten
123          strip.setPixelColor(uint8_t(24), strip.Color(0, 0, 0));
124          strip.setPixelColor(uint8_t(32), strip.Color(0, 0, 0));
125          strip.setPixelColor(uint8_t(18), strip.Color(0, 0, 0));
126          strip.setPixelColor(uint8_t(21), strip.Color(0, 0, 0));
127          strip.setPixelColor(uint8_t(50), strip.Color(0, 0, 0));
128          strip.setPixelColor(uint8_t(51), strip.Color(0, 0, 0));
129          strip.setPixelColor(uint8_t(52), strip.Color(0, 0, 0));
130          strip.setPixelColor(uint8_t(53), strip.Color(0, 0, 0));
131          strip.setPixelColor(uint8_t(39), strip.Color(0, 0, 0));
132
133          strip.show();
134          delay(15);
135      }
136 }
137 //=============8x8Matrix========
138
139 void loop()
140 {
141     // establish variables for duration of the ping, and the distance result
142     // in inches and centimeters:
143     long duration, inches, cm;
144
145     // The PING))) is triggered by a HIGH pulse of 2 or more microseconds.
146     // Give a short LOW pulse beforehand to ensure a clean HIGH pulse:
147     pinMode(pingPin, OUTPUT);
148     //
149     digitalWrite(pingPin, LOW);
150     delayMicroseconds(2);
151     digitalWrite(pingPin, HIGH);
152     delayMicroseconds(5);
153     digitalWrite(pingPin, LOW);
154
155     // The same pin is used to read the signal from the PING))): a HIGH pulse
156     // whose duration is the time (in microseconds) from the sending of the
            ping
157     // to the reception of its echo off of an object.
158     pinMode(pingPin, INPUT);
159     duration = pulseIn(pingPin, HIGH);
160
161     // convert the time into a distance
162     inches = microsecondsToInches(duration);
163     cm = microsecondsToCentimeters(duration);
164
```

```csharp
165        //Serial.print(inches);
166        //Serial.print("in, ");
167        Serial.print(cm);
168        //Serial.print("cm");
169        Serial.println();
170
171        delay(100);
172        if (cm < Abstand) // wenn Abstand kleiner als der oben eingetragene Wert ⏎
             dann
173        {
174            //Serial.print("zu dicht");
175            digitalWrite(Piep, HIGH); // Soll den Störenfried signalisieren und er ⏎
                 schaut zu dem Piep hin und sieht den Smiley
176            bad(strip.Color(0, 15, 0));
177            //rote LEDs für traurig,
178            //Hinweis: je höher der Wert um so heller leuchten die LEDs und mehr ⏎
                 Strom ziehen sie.
179            //Bei diesem Werten liegt der Strombedarf bei ca. 100 mA. Ist aber ⏎
                 auch abhängig von der Anzahl der LED's
180            strip.show();
181        }
182        else
183        {
184            digitalWrite(Piep, LOW);
185            smiley(strip.Color(0, 0, 25)); //Blau für lachen
186            strip.show();
187
188        }
189 }
190
191 long microsecondsToInches(long microseconds)
192 {
193     // According to Parallax's datasheet for the PING))), there are 73.746
194     // microseconds per inch (i.e. sound travels at 1130 feet per second).
195     // This gives the distance travelled by the ping, outbound and return,
196     // so we divide by 2 to get the distance of the obstacle.
197     // See: http://www.parallax.com/dl/docs/prod/acc/28015-PING-v1.3.pdf
198     return microseconds / 74 / 2;
199 }
200
201 long microsecondsToCentimeters(long microseconds)
202 {
203     // The speed of sound is 340 m/s or 29 microseconds per centimeter.
204     // The ping travels out and back, so to find the distance of the object we
205     // take half of the distance travelled.
206     return microseconds / 29 / 2;
207 }
```