Ultra Sonic range measurement module

Seeed ultrasonic sensor is non-contact distance measurement module, which is also compatible with electronic brick. It's designed for easy modular project usage with industrial performance.

Features

- Detecting range: 3cm-4m
- Best in 30 degree angle
- Electronic brick compatible interface
- 5VDC power supply
- Breadboard friendly
- Dual transducer
- Arduino library ready

Specification

Supply voltage	5V
Global Current Consumption	15 mA
Ultrasonic Frequency	40k Hz
Maximal Range	400 cm
Minimal Range	3 cm
Resolution	1 cm
Trigger Pulse Width	10 µs
Outline Dimension	43x20x15 mm

Getting started

A short ultrasonic pulse is transmitted at the time 0, reflected by an object. The sensor receives this signal and converts it to an electric signal. The next pulse can be transmitted when the echo is faded away. This time period is called cycle period. The recommend cycle period should be no less than 50ms. If a 10µs width trigger pulse is sent to the signal pin, the Ultrasonic module will output eight 40kHz ultrasonic signal and detect the echo back. The measured distance is proportional to the echo pulse width and can be calculated by the formula above. If no obstacle is detected, the output pin will give a 38ms high level signal.

Play with Arduino

Hardware

- **Step 1.** Prepare the below stuffs:

Seeeduino V4.2	Base Shield	Ultra_Sonic_range_measurement_module

- **Step 2.** Connect Ultra_Sonic_range_measurement_module to port **D2** of Grove-Base Shield.

- **Step 3.** Plug Grove - Base Shield into Seeeduino.

- **Step 4.** Connect Seeeduino to PC via a USB cable.

Note

If we don't have Grove Base Shield, We also can directly connect this module to Seeeduino as below.

Seeeduino	???
5V	Red
GND	Black
Not Conencted	White
D2	Yellow

Software

Step 1. Copy the code and flash it into the controller board. **Step 2.** Upload the code and open monitor window.

```
1  #include "Arduino.h"
2  class Ultrasonic
3  {
4     public:
5     Ultrasonic(int pin);
6     void DistanceMeasure(void);
7     double microsecondsToCentimeters(void);
8     double microsecondsToInches(void);
9     private:
10    int this_pin;//pin number of Arduino that is connected with SIG pin of Ultrasonic Ranger.
11    long duration;// the Pulse time received;
12  };
13  Ultrasonic::Ultrasonic(int pin)
14  {
15    this_pin = pin;
16  }
17  /*Begin the detection and get the pulse back signal*/
18  void Ultrasonic::DistanceMeasure(void)
19  {
20    pinMode(this_pin, OUTPUT);
21    digitalWrite(this_pin, LOW);
22    delayMicroseconds(2);
23    digitalWrite(this_pin, HIGH);
24    delayMicroseconds(5);
25    digitalWrite(this_pin,LOW);
26    pinMode(this_pin,INPUT);
```

```
27    duration = pulseIn(this_pin,HIGH);
28}
29/*The measured distance from the range 0 to 400 Centimeters*/
30double Ultrasonic::microsecondsToCentimeters(void)
31{
32    return duration/29.0/2.0;
33}
34/*The measured distance from the range 0 to 157 Inches*/
35double Ultrasonic::microsecondsToInches(void)
36{
37    return duration/74.0/2.0;
38}
39
40Ultrasonic ultrasonic(2);
41void setup()
42{
43    Serial.begin(9600);
44}
45void loop()
46{
47    double RangeInInches;
48    double RangeInCentimeters;
49    ultrasonic.DistanceMeasure();// get the current signal time;
50    RangeInInches = ultrasonic.microsecondsToInches();//convert the time to inches;
51    RangeInCentimeters = ultrasonic.microsecondsToCentimeters();//convert the time to centimeters
52    Serial.println("The distance to obstacles in front is: ");
53    Serial.print(RangeInInches);//0~157 inches
54    Serial.println(" inch");
55    Serial.print(RangeInCentimeters);//0~400cm
56    Serial.println(" cm");
57    delay(1000);
58}
```

Tech Support

Please submit any technical issue into our forum or drop mail to techsupport@seeed.cc.