

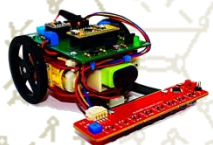
Ультразвуков сензор HC-SR04

Разработил: инж.А.Анчев



1.Общи сведения

- ❗ **Ехолокация** е процес, който се състои в изпращането на сигнали с различна честота (радиовълни, ултразвук, звук) и приемане на отразената вълна.
- ❗ Използва се с цел откриване на предмети и тела, като положението им се определя по времето между изпратения и получения сигнал.



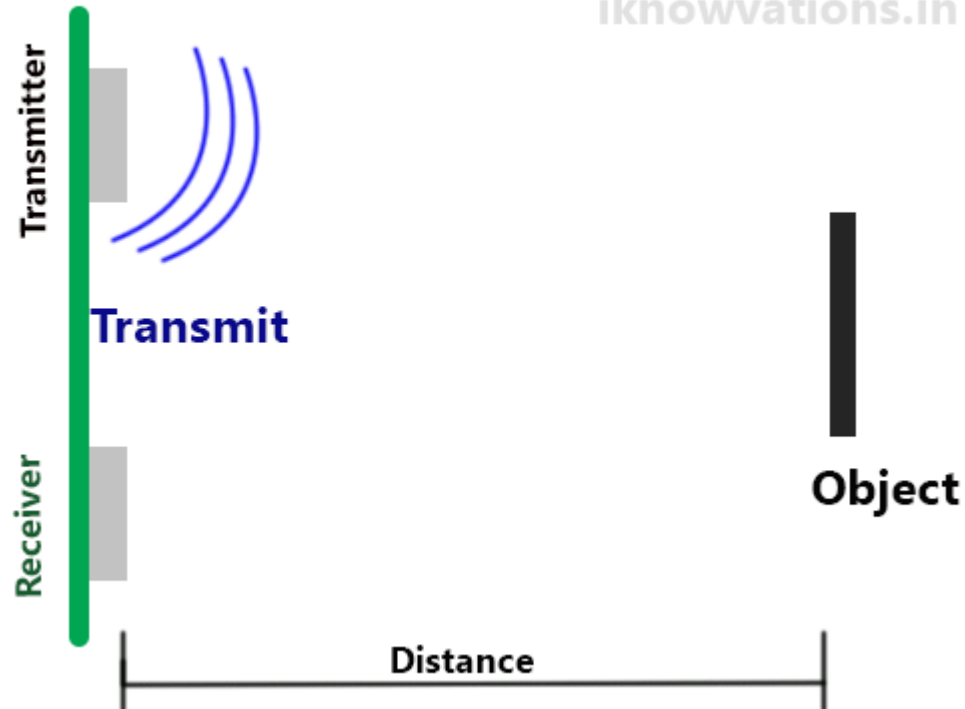


Ултразвуков сензор HC-SR04

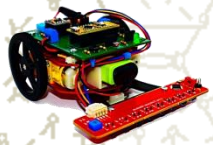
- Ултразвуковият сензор **HC-SR04** използва звукови вълни с честота 40kHz;

Working of Ultrasonic Sensor module

iknowvations.in



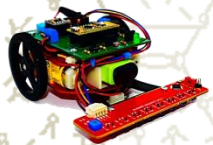
SOCIETY
ROBOTICS



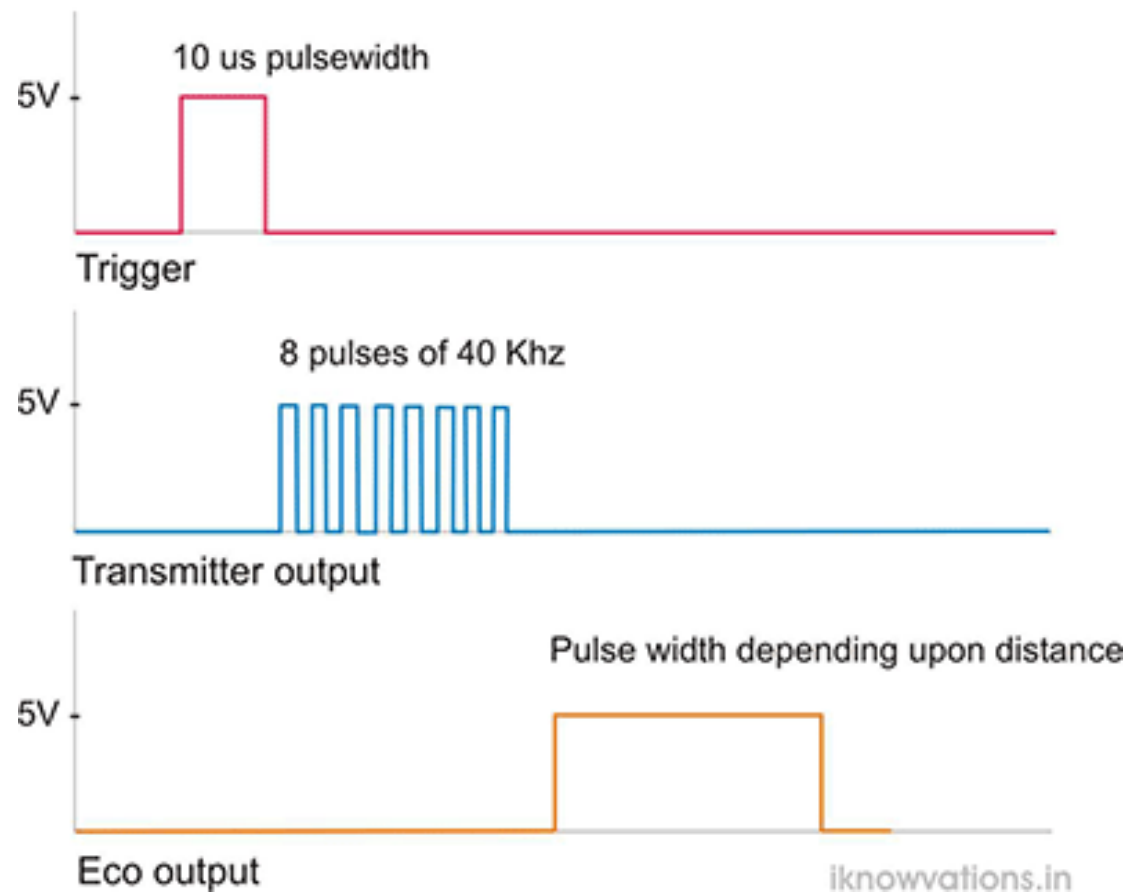


Society of Robotics

SOCIETY OF ROBOTICS

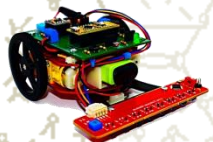


Ultrasonic sensor HC-SR04 working





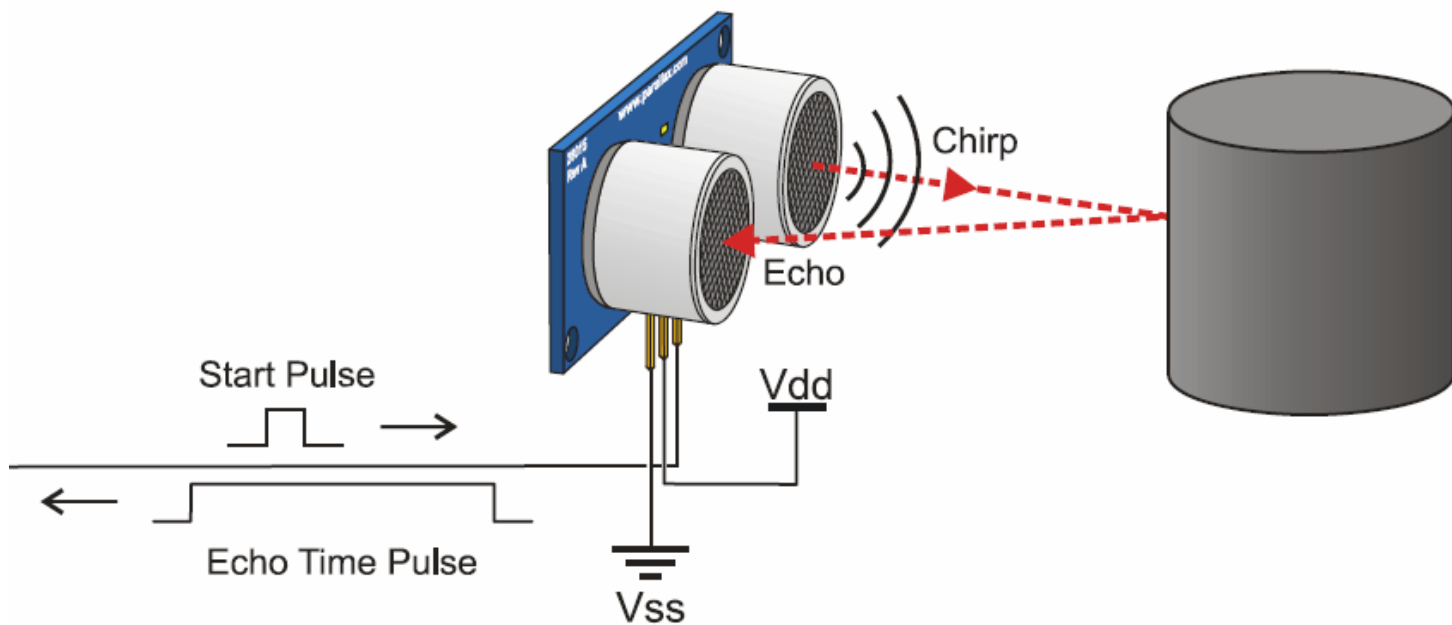
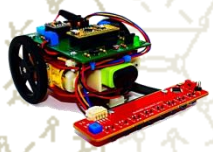
- ✱ Скоростта на звука е разстоянието, изминато от звукова вълна за единица време. При 20°C скоростта на звука е **343 м в секунда**;
- ✱ Тъй като ние ще използваме тези сензори за близко разстояние, ние ще го превърнем в **0.034 см за 1 микросекунда**;
- ✱ Времето, което сме измерили, трябва да разделим на 2, защото в него влиза пътят на звуковата вълна до предмета и обратно;





Abirah

SOCIETY
ROBOTIC

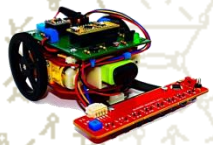


разстояние = скорост на звука * време / 2



Abhishek

SOCIETY
ROBOTIC--



+5V

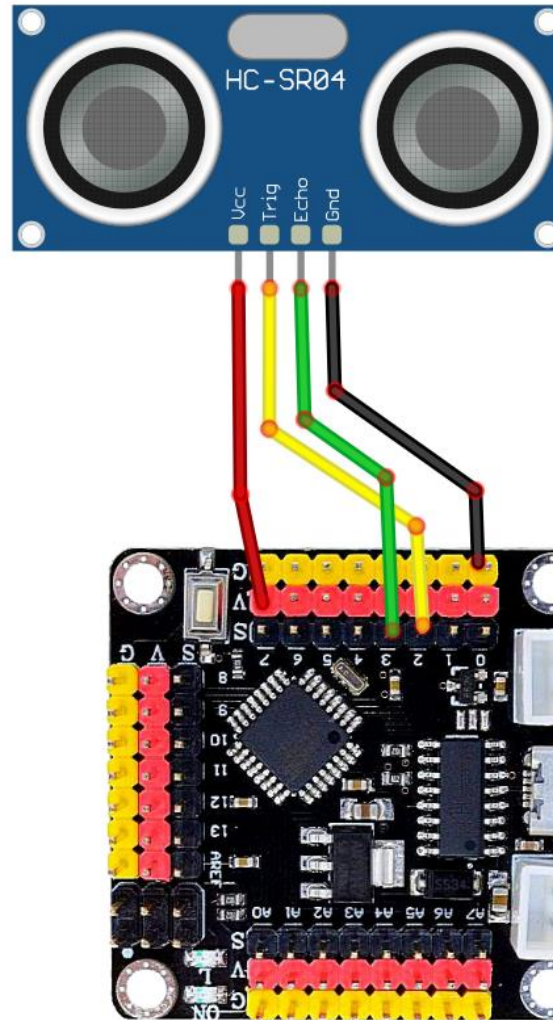
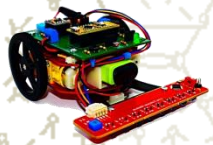
Dig Pin

Dig Pin

GND



**SOCIETY
ROBOTIC**



HC-SR04

NANO Strong



Характеристики

Захр.напрежение: 5V

Консумация на ток: 6mA

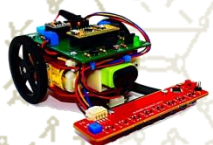
Ултразв.честота: 40kHz

Обхват: ~ 2cm - 450cm

Работно напрежение: 5V

Ъгъл на отчитане: 30°

SOCIETY
ROBOTICS





Програмен код

```
5
6 | HC-SR04 | Arduino | | 3 pins | Arduino |
7
8 | Vcc | 5V | | Vcc | 5V |
9 | Trig | 2 | OR | SIG | 13 |
10 | Echo | 3 | | Gnd | GND |
11 | Gnd | GND | -----
12
13 */
14 // ----- VARIABLES
15 const int trigPin = 2;
16 const int echoPin = 3;
17 String label = "Distance :";
18 String label1 = " cms. ";
19 long duration;
20 int distance;
21
22 // ===== SETUP =====
23 void setup() {
24   pinMode(trigPin, OUTPUT);
25   pinMode(echoPin, INPUT);
26   Serial.begin(9600);
27 }
28
29 // ===== LOOP =====
30 void loop() {
31   // Поставяме trigPin в състояние LOW, за да го подготвим за излъчване на импулс
32   digitalWrite(trigPin, LOW);
33   delayMicroseconds(5);
34
35   // ----- Сега поставяме trigPin в състояние HIGH за 10 микросекунди, което ще накара датчика да излъчи импулс
36   digitalWrite(trigPin, HIGH);
37   delayMicroseconds(10);
38   digitalWrite(trigPin, LOW);
39
40   // ----- Поставяме echoPin в състояние HIGH, за да приемем отразения импулс и измерваме времето за приемане на отразения импулс в микросекунди
41   duration = pulseIn(echoPin, HIGH);
42
43   // ----- Изчисляваме разстоянието до препятствието в см на базата на времето на отразения импулс
44   distance = duration * 0.034 / 2;
45
46   // ----- Отпечатваме стойност на полученото разстояние на серийния монитор
47   Serial.println(label + distance + label1);
48   delay(70);
49 }
```

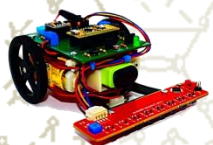
SOCIETY
ROBOTIC





Използвани методи

- ✦ `delayMicroseconds(t)` - поставя изпълнението на програмата на пауза за време `t` в микросекунди;
- ✦ `pulseIn(pin, value)` – Отчита времето между промяната на състоянието на даден пин (`pin`). Връща дължината на импулса в микросекунди;

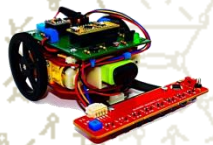




Работа с библиотеки

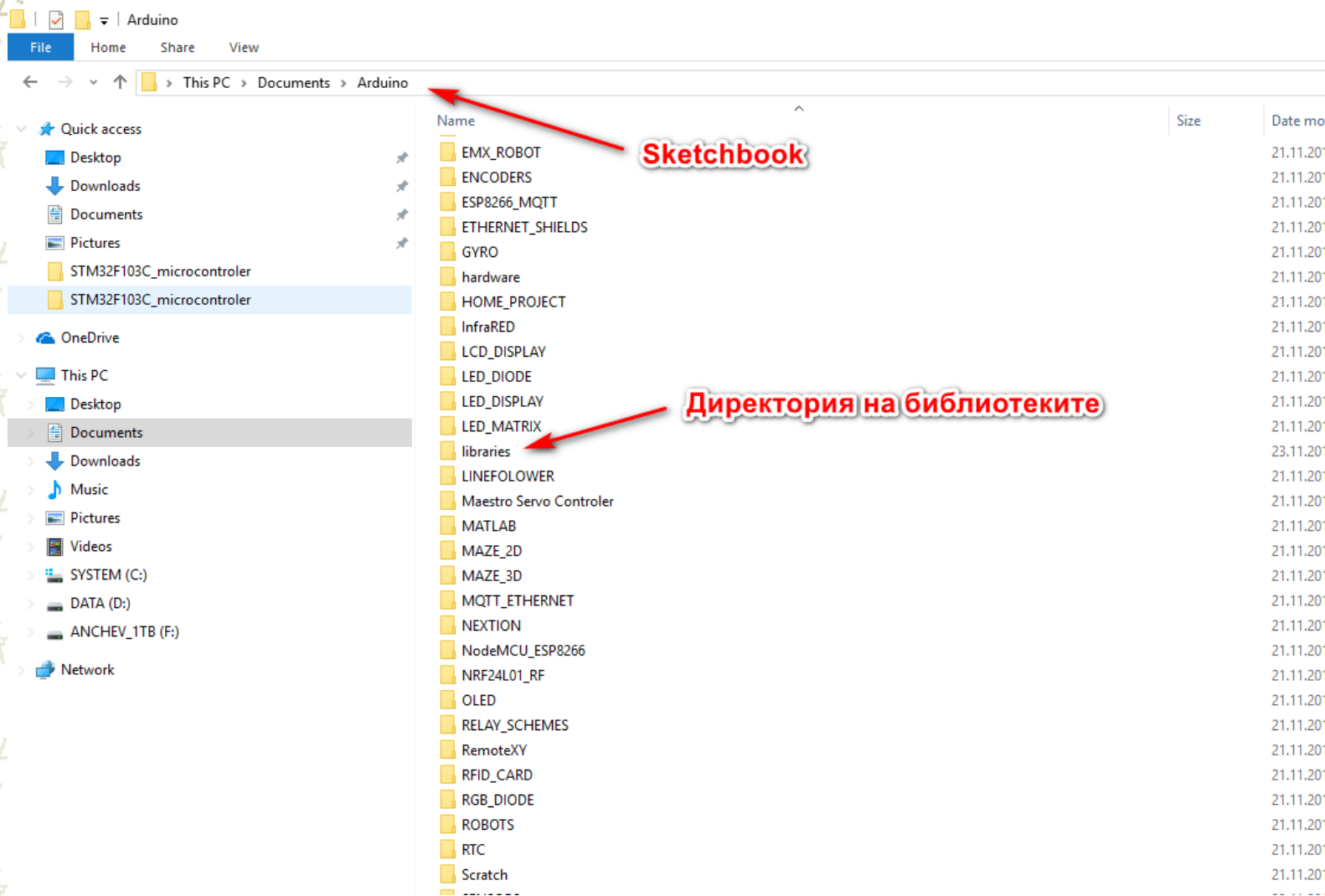
- ✱ Библиотеката е набор от функции, предназначени да направят работата с различни сензори, LCD екрани, модули и т.н. възможно най-лесни.
- ✱ Най-често библиотеките са подредени под формата на ZIP архив или само папка.
- ✱ Името на тази папка е името на библиотеката. Вътре в папката има файл с разширение **.cpp**, файл с разширение **.h**, както и текстов файл с име **keywords.txt**, папка с примери и други файлове, изисквани от библиотеката.

SOCIETY
ROBOTICS





Местоположение





Структура

File Explorer window showing the structure of the Ultrasonic library.

Path: This PC > Documents > Arduino > libraries > Ultrasonic

Left sidebar (Quick access):

- Desktop
- Downloads
- Documents
- Pictures
- STM32F103C_microcontroler
- STM32F103C_microcontroler
- OneDrive
- This PC
- Desktop
- Documents
- Downloads

Right pane (Files and folders):

- examples (Folder) ← **Примери**
- extras (Folder)
- src (Folder) ← **Файлове на библиотеката**
- keywords (File)
- library.properties (File)

Bottom pane (Files):

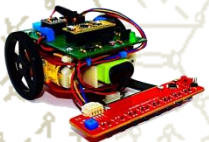
Name	Size	Type	Date modified
Ultrasonic.cpp	2 KB	CPP File	
Ultrasonic.h	2 KB	H File	



.h файл

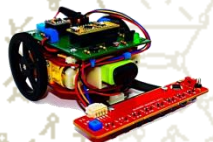
```
1 #ifndef Ultrasonic_h
2 #define Ultrasonic_h
3
4 /*
5  * Values of divisors
6  */
7 #define CM 28
8 #define INC 71
9
10 class Ultrasonic {
11 public:
12     Ultrasonic(uint8_t sigPin) : Ultrasonic(sigPin, sigPin) {};
13     Ultrasonic(uint8_t trigPin, uint8_t echoPin, unsigned long timeOut = 20000UL);
14     unsigned int read(uint8_t und = CM);
15     unsigned int distanceRead(uint8_t und = CM) __attribute__((deprecated ("This method is deprecated, use read() instead.")));
16     void setTimeout(unsigned long timeOut) {timeout = timeOut;}
17
18 private:
19     uint8_t trig;
20     uint8_t echo;
21     boolean threePins = false;
22     unsigned long previousMicros;
23     unsigned long timeout;
24     unsigned int timing();
25 };
26
27 #endif // Ultrasonic_h
```

Потребителски функции





SOCIETY
ROBOTIC



.cpp файл

```
1 #if ARDUINO >= 100
2   #include <Arduino.h>
3 #else
4   #include <WProgram.h>
5 #endif
6
7 #include "Ultrasonic.h"
8
9 Ultrasonic::Ultrasonic(uint8_t trigPin, uint8_t echoPin, unsigned long timeout) {
10   trig = trigPin;
11   echo = echoPin;
12   threePins = trig == echo ? true : false;
13   pinMode(trig, OUTPUT);
14   pinMode(echo, INPUT);
15   timeout = timeout;
16 }
17
18 unsigned int Ultrasonic::timing() {
19   if (threePins)
20     pinMode(trig, OUTPUT);
21
22   digitalWrite(trig, LOW);
23   delayMicroseconds(2);
24   digitalWrite(trig, HIGH);
25   delayMicroseconds(10);
26   digitalWrite(trig, LOW);
27
28   if (threePins)
29     pinMode(trig, INPUT);
30
31   previousMicros = micros();
32   while(!digitalRead(echo) && (micros() - previousMicros) <= timeout); // wait for the echo pin HIGH or timeout
33   previousMicros = micros();
34   while(digitalRead(echo) && (micros() - previousMicros) <= timeout); // wait for the echo pin LOW or timeout
35
36   return micros() - previousMicros; // duration
37 }
38
39 /*
40  * If the unit of measure is not passed as a parameter,
41  * sby default, it will return the distance in centimeters.
42  * To change the default, replace CM by INC.
43  */
44 unsigned int Ultrasonic::read(uint8_t und) {
45   return timing() / und / 2; //distance by divisor
46 }
47
48 /*
49  * This method is too verbal, so, it's deprecated.
50  * Use read() instead.
51  */
52 unsigned int Ultrasonic::distanceRead(uint8_t und) {
53   return read(und);
54 }
55 }
```

Управление





Програмен код

```
1 /*
2  Arduino NANO with HC-SR04 ultrasonic
3  Byala Robots Club
4
```

5	-----			-----		
6	HC-SR04	Arduino		3 pins	Arduino	
7	-----			-----		
8	Vcc	5V		Vcc	5V	
9	Trig	2	OR	SIG	13	
10	Echo	3		Gnd	GND	
11	Gnd	GND		-----		
12						

```
13 */
14 #include <Ultrasonic.h>
15 Ultrasonic sensor(2, 3);
16
```

конструктор

```
17 void setup() {
18   Serial.begin(9600);
19 }
```

Функция за измерване

```
20
21 void loop() {
22   int distance = sensor.distanceRead();
23   Serial.println(distance);
24   delay(100);
25 }
```

**SOCIETY
ROBOTIC--**





Добавяне на библиотека

HC_SR04_ultrasonic_library | Ардуино 1.8.7

Файл Редактиране Скица Инструменти Помощ

✓ ↩ ⏏ ⚙

HC_SR04_ultras

```
1 /*
2   Arduino
3   Byala Ro
4
5   -----
6   | HC-SR04
7
8   | Vcc | 5V | | Vcc | 5V |
9   | Trig | 12 | OR | SIG | 13 |
10  | Echo | 13 | | Gnd | GND |
11  | Gnd | GND | |
12
13 */
14 #include <Ultrasonic.h>
15 Ultrasonic sensor(12, 13);
16
17 void setup() {
18   Serial.begin(9600);
19 }
20
21 void loop() {
22   int distance = sensor.distanceRead();
23   Serial.println(distance);
24   delay(100);
25 }
26
27
```

Проверка/компилиране Ctrl+R
Качване Ctrl+U
Качване чрез програматор Ctrl+Shift+U
Изнасяне на компилирания двоичен файл Ctrl+Alt+S
Показване на папката на скицата Ctrl+K
Включване на библиотека
Добавяне на файл

SimpleDHT
SimpleRGB
SoftI2CMaster
SparkFun Si7021 Humidity and Temperature Sensor
SparkFun TSL2561
SparkFun BMP180
Suli
SunFounder Line Follower
TFTLCD
TFTv2
TM1637
TM1637Display
TSL2561
ThingSpeak
Ticker
Time
Timer
TimerOne
TimerOne-v9
Tlc5940
Tone_ltead
Touch-Screen
U8g2
U8glib
UIPEthernet
UTFT
UTFT_Butttons
UTFT_Geometry
UTFT_SPIflash
UTFT_tinyFAT
UTouch
Ucglib
Ultrasonic
VarSpeedServo
WEMOS Matrix LED Shield
WEMOS SHT3x

Избираме библиотеката

SOCIETY
ROBOTIC

