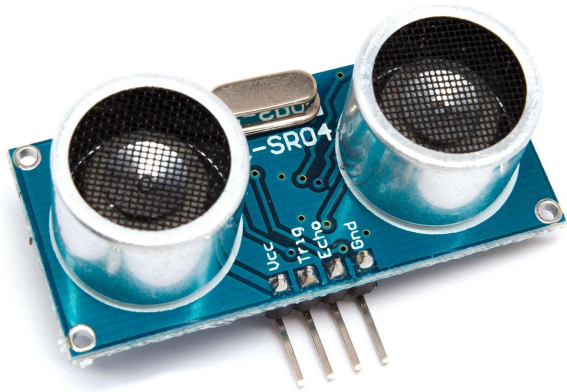
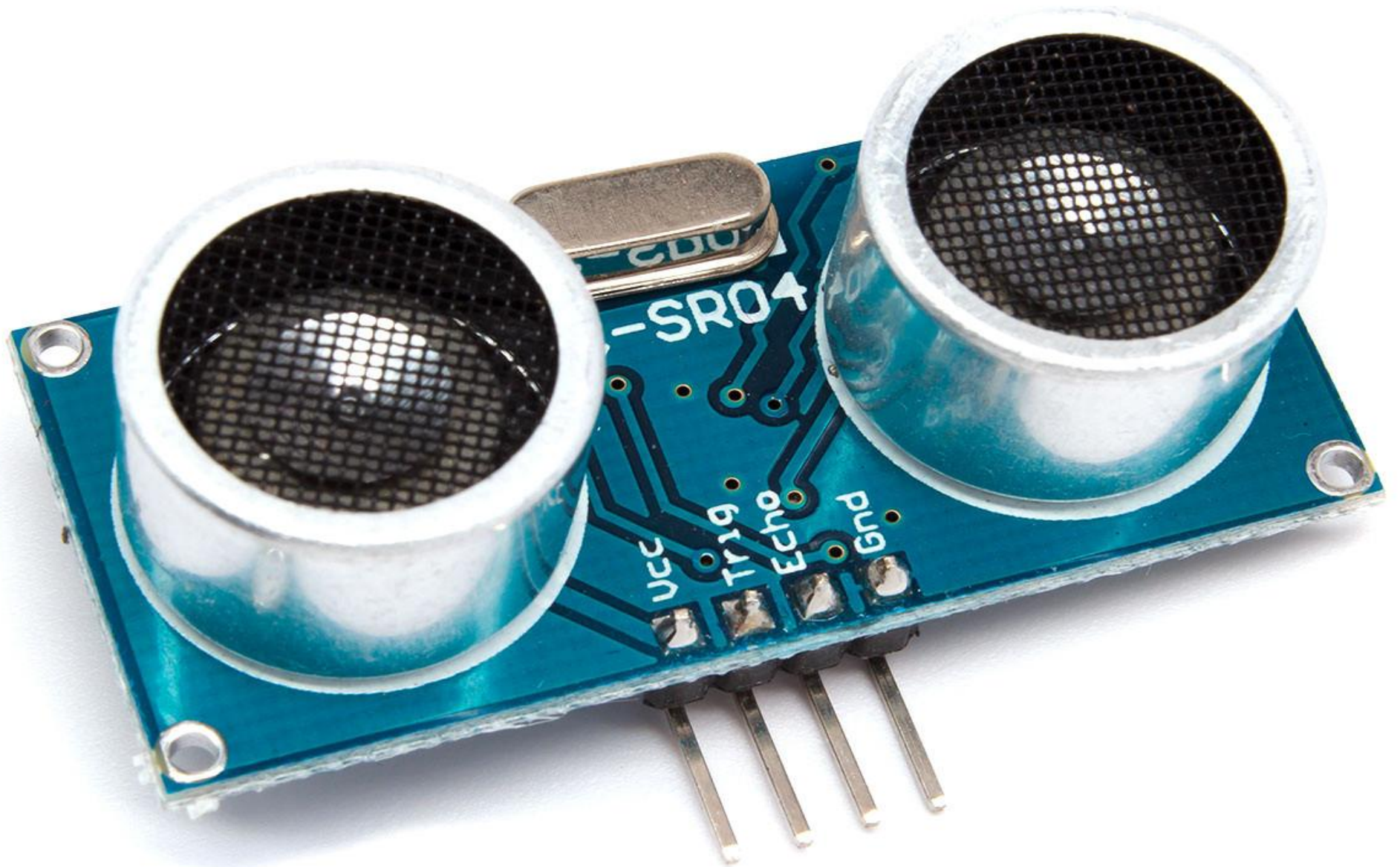


# Ультразвуковой дальномер

# Ультразвуковой дальномер



Ультразвуковой дальномер определяет расстояние до объектов точно так же, как это делают дельфины или летучие мыши. Он генерирует звуковые импульсы на частоте 40 кГц и слушает эхо. По времени распространения звуковой волны туда и обратно можно однозначно определить расстояние до объекта. В отличие от инфракрасных дальномеров, на показания



# Пример работы УЗ-дальномера

Определяем пины:

```
#define TRIG_PIN 7  
#define ECHO_PIN 8
```

**setup()**

Настройка контактов TRIG и ECHO:

```
pinMode(TRIG_PIN, OUTPUT);  
pinMode(ECHO_PIN, INPUT);
```

Устанавливаем скорость соединения с последовательным портом:

```
Serial.begin(9600);
```

**loop()**

Генерируем звуковую волну:

```
digitalWrite(TRIG_PIN, HIGH);  
digitalWrite(TRIG_PIN, LOW);
```

# Пример работы УЗ-дальномера

Принимаем звуковую волну и определяем расстояние:

```
int dist = pulseIn(ECHO_PIN, HIGH) / 50;
```

Коэффициент 50 нужен для перевода в сантиметры.

Печать в последовательный порт:

```
Serial.println(dist);
```

Третий параметр ограничивает время ожидания (измеряем расстояние не более 3-х метров):

```
int dist = pulseIn(ECHO_PIN, HIGH, 15000) / 50;
```

Ограничиваем значение dist от 1 до 300:

```
dist = constrain(dist, 1, 300);
```

# Пьезодинамик



# Пример работы пьезодинамика

В динамике уже есть резистор.

Определяем пины:

```
#define ZUMMER 10
```

`setup()`

Настройка пина:

```
pinMode(ZUMMER, OUTPUT);
```

`loop()`

Использование динамика:

```
tone(ZUMMER, 70, 10);
```

ZUMMER — пин к которому подключен динамик;

70 — частота звука в Гц;

10 — время работы в миллисекундах