

# Модуль чувствительного микрофонного датчика (Big Sound)

## Sensitive microphone sensor module KY-037



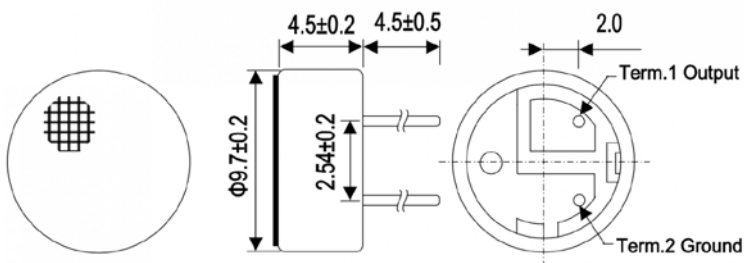
### Краткое описание KY-037

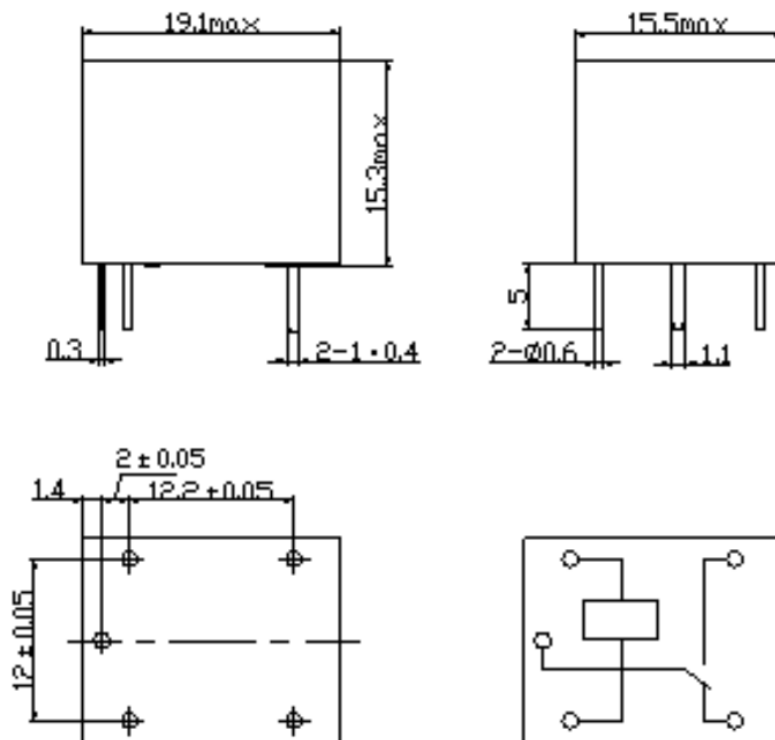
Датчик имеет два выхода: аналоговый и цифровой. Сигнал на аналоговом выходе пропорционален уровню шума вокруг датчика. Чем выше громкость – тем ниже значение напряжения на нём.

Сигнал на цифровом выходе появляется лишь при превышении заданного порога громкости (это происходит при помощи встроенного компаратора), всё остальное время на нём логический ноль. При появлении сигнала на выходе загорится светодиод L2. Светодиод L1 является индикатором питания.

Чувствительность датчика настраивается с помощью находящегося на плате многооборотного подстроечного резистора. С его помощью это получается довольно точно. Регулируется и уровень аналогового сигнала, и порог срабатывания компаратора.

Датчик работает при напряжении от 3.3 до 5 В, благодаря чему совместим с большинством микроконтроллеров.





Контакт	Назначение
A0	Напряжение соответствует уровню громкости окружающих шумов.
G	Общий провод.
+	Питание положительный полюс.
DO	Выход логического сигнала сообщаемого о превышении установленного порога.

## ОСНОВНЫЕ ДОСТОИНСТВА

- Простота использования

- Наличие двух выходов различных типов
- Тонкая настройка чувствительности
- Светодиодные индикаторы

## ПРИМЕРЫ ИСПОЛЬЗОВАНИЯ

- Выключатели со срабатыванием по хлопку
- Измеритель уровня громкости (VU meter)

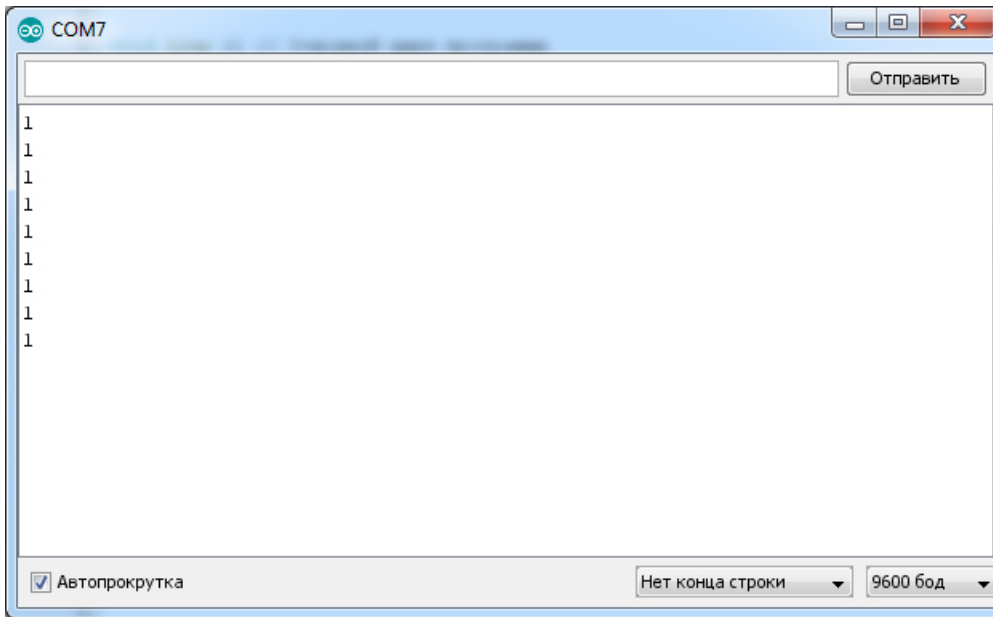
### Настройка датчика звука KY-037.

Подключаем плату Arduino Nano к компьютеру. На модуле датчика хлопков **KY-037**, должен сразу загореться индикатор питания **L1**. Необходимо сначала взять отвертку и подкрутить подстроечный резистор, настроив тем самым чувствительность датчика. А в настройке чувствительности нам поможет индикатор срабатывания датчика **L2**. Если индикатор **L2** при включении модуля тоже загорается, крутим подстроечный резистор против часовой стрелки до тех пор, пока не дойдем до момента затухания индикатора. Если же индикатор **L2** находится в выключенном состоянии при включении модуля, значит наоборот, крутим подстроечный резистор по часовой стрелке, пока не дойдем до момента, когда индикатор начнет загораться. В итоге в этом месте, где чуть повернув подстроечный резистор в одну или другую сторону, индикатор стремится потухнуть или загореться, нам нужно повернуть совсем немного против часовой стрелки, чтобы индикатор **L2** потух, но при хлопках в ладоши пытался загораться.

Открываем программу Arduino IDE, создаем новый файл и вставляем в него код, который нам покажет каким образом приходит цифровой сигнал с вывода **D0** в случаях превышения порога шума установленного с помощью подстроечного резистора.

```
01.const int sensorD0 = 5; // Пин Arduino к которому подключен пин D0 датчика
02.
03.void setup () // Настройки
04. {
05.   Serial.begin (9600); // Инициализация SerialPort
06. }
07.
08.void loop () // Основной цикл программы
09. {
10.   int sensorValue = digitalRead(sensorD0); // получаем сигнал от датчика
11.   if (sensorValue == true) // Если пришел сигнал высокого уровня
12.     Serial.println(sensorValue); // Выводим цифровое значение на терминал
13. }
```

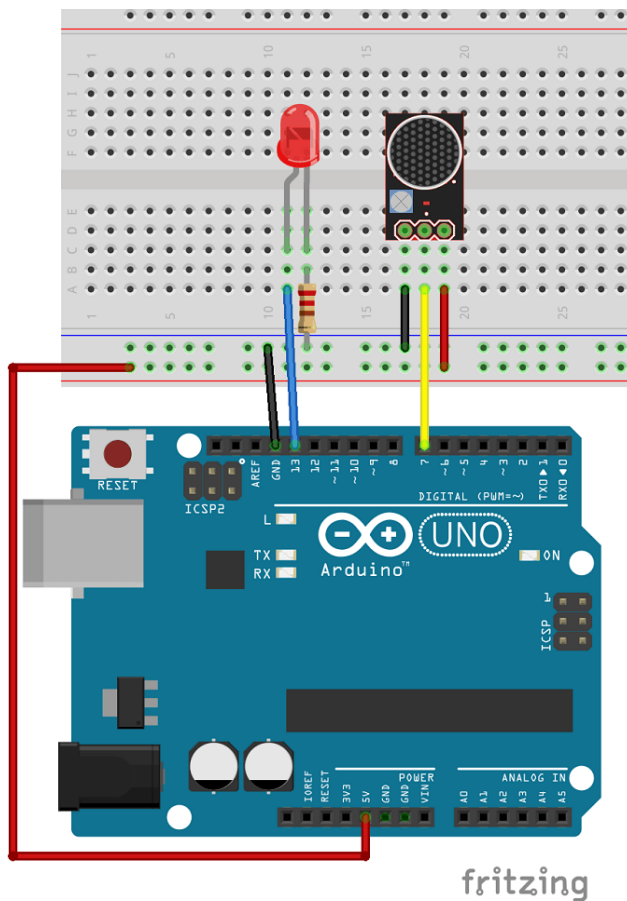
Заливаем данный скетч и переходим в меню "**Инструменты**" - "**Монитор порта**". Окно мониторинга порта будет пустым, но как только мы будем хлопать в ладоши, в окне появятся единицы, говорящие о наличии сигнала высокого уровня на выводе D0 модуля датчика звука.



Всё хорошо. Мы настроили датчик и убедились, что наша Ардуино прекрасно принимает сигнал от него.

### Включаем свет по хлопку и выключаем автоматически по таймеру.

Теперь добавим в нашу схему обычный светодиод и напишем простой код, который будет при обнаружении шума зажигать светодиод и тушить его по истечении какого-то времени.



## Датчик звука KY-037 включает свет по хлопку и выключает свет по хлопку.

Загрузим новый скетч, который по хлопку будет включать или выключать наш светодиод. Светодиод мы взяли для примера, нет ни каких проблем подсоединить вместо него модуль реле и тем самым включать или выключать любые бытовые приборы.

```
01.const int sensorD0 = 5; // Пин Arduino к которому подключен выход D0 датчика
02.const int diod = 2; // Пин Arduino к которому подключен светодиод
03.int diodState = LOW; // Статус светодиода "выключен"
04.
05.void setup ()
06. {
07.  pinMode(diod, OUTPUT); // устанавливаем цифровой пин 2 в режим выхода
08. }
09.
10.void loop ()
11. {
12.  int sensorValue = digitalRead(sensorD0); // получаем сигнал с датчика
13.
14.  if (sensorValue == 1 && diodState == LOW) //если порог громкости достигнут и светодиод был ВЫКЛЮЧЕН
15.  {
16.    digitalWrite(diod, HIGH); // включаем светодиод
17.    diodState = HIGH; // устанавливаем статус светодиода "включен"
18.    delay(100); // небольшая задержка для фильтрации помех
19.  }
20.  else // иначе
21.  {
22.    if (sensorValue == 1 && diodState == HIGH) // если порог громкости достигнут и светодиод был ВКЛЮЧЕН
23.    {
24.      digitalWrite(diod, LOW); // выключаем светодиод
25.      diodState = LOW; // устанавливаем статус светодиода "выключен"
26.      delay(100); // небольшая задержка для фильтрации помех
27.    }
28.  }
29. }
```

Хлопаем теперь один раз в ладоши, свет загорается. Хлопаем повторно в ладоши, светодиод тухнет.