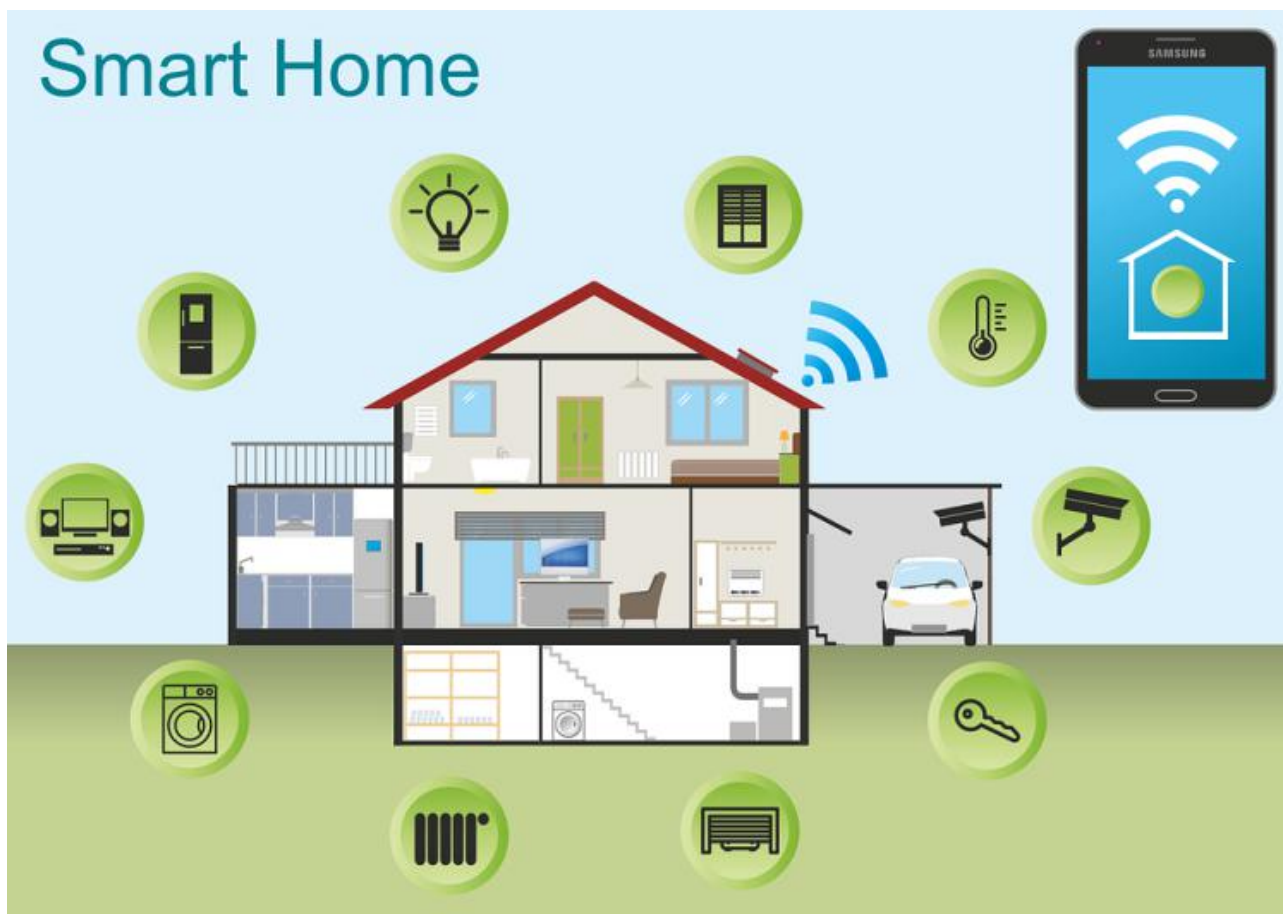




Комплект умного дома Keyestudio для Arduino





Содержание

1. Обзор:	3
2. Комплект	4
3. Загрузите программное обеспечение и установите драйвер	9
(1) Загрузить программное обеспечение	9
(2) Плата разработки Keyestudio PLUS	11
(3) Установка драйвера	13
(4) Настройка Arduino IDE	19
(4) Запустите вашу первую программу	24
4. Как добавить библиотеку?	27
5. Проекты	33
Проект 1: мигание светодиода	34
Проект 2: Дыхательный свет	41
Проект 3: Пассивный зуммер	49
Проект 4: Датчик кнопки	62
Проект 5: 1-канальный релейный модуль	66
Проект 6: Датчик фотоэлемента	71
Проект 7: Регулировка угла сервопривода двигателя	76
Проект 8: Модуль вентилятора	81
Проект 9: Датчик пара	85
Проект 10: Датчик движения PIR	90
Проект 11: Аналоговый газ (MQ-2) Датчик	96



Проект 12: 1602 ЖК-дисплей	102
Проект 13: Датчик влажности почвы	107
Проект 14: Тест Bluetooth	113
Использование приложения Bluetooth	116
6. Руководство по сборке	126
Проект 15: Многоцелевой умный дом	159
7. Связанные ресурсы	190

1. Обзор:



Этот учебный комплект для умного дома, основанный на платформе Arduino, был недавно выпущен компанией Keyestudio DIY Robot Co. Ltd.

Он имитирует настоящий умный дом и демонстрирует людям уютную и комфортную жизнь.



Фактически, логическое программирование, невидимая рука, контролирует все в умном доме: она включает кондиционер, включает водонагреватель, защищает ваш дом электронным замком и настраивает ваши светодиодные фонари и умные шторы на автоматическое включение. когда придешь домой. Между тем, интеллектуальная система освещения позволяет создать комфортную, спокойную атмосферу. Все завершается с помощью пульта дистанционного управления или вашего собственного мобильного телефона.

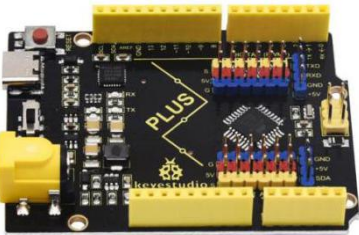
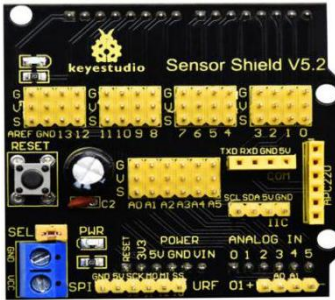


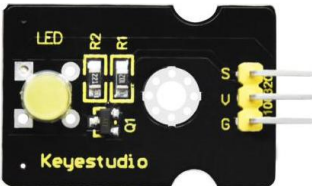

Как выразился Билл Гейтс: «В ближайшем будущем дом без системы умного дома будет столь же немодным, как дом без доступа к Интернету сегодня».

Итак, вперед и приступайте; давайте построим этот удивительный аналоговый умный дом.


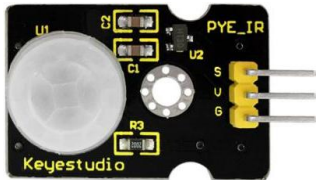


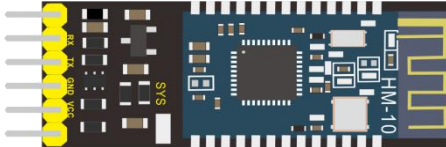
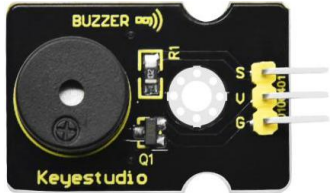
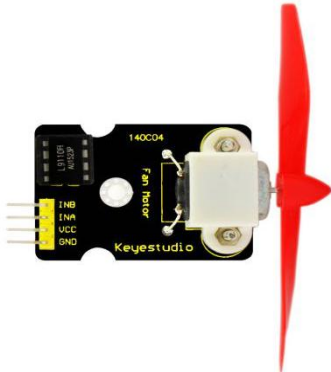
2. Комплект

Получив этот комплект для умного дома, нам нужно убедиться, что в нем нет недостающих компонентов.

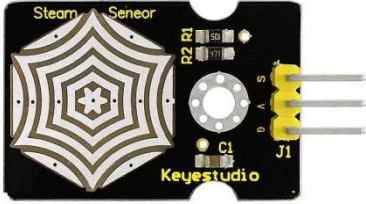







#	Имя	КОЛИЧЕСТВО	Картина
1	Панель управления Keyestudio PLUS (Совместим с Arduino ООН)	1	
2	Экран сенсора Keyestudio V 5.2	1	
3	Деревянная доска * 10 т = 3 мм	1	
4	Белый светодиодный модуль	1	
5	Желтый светодиодный модуль	1	
6	Датчик кнопки	2	














7	Датчик фотозлемента	1	
8	Датчик движения PIR	1	
9	Датчик газа MQ-2	1	
10	Модуль реле	1	
11	Модуль Bluetooth HM-10	1	
12	Пассивный датчик зуммера	1	
13	Модуль вентилятора	1	



14	Датчик пара	1	
15	Серводвигатель	2	
16	Модуль дисплея LCD1602	1	
17	Датчик влажности почвы	1	
18	USB-кабель	1	
19	Кабели Dupont между женщинами и женщинами	40	
20	Кабели Dupont Cables между мужчинами и женщинами	6	
21 год	Никелированные гайки M3	25	
22	Винты с полукруглой головкой M2 * 12 мм	6	
23	Никелированные гайки M2	6	
24	M3 * 10 мм, двухходовые	4	



	Медный куст		
25	Винты с полукруглой головкой M3 * 6 мм	8	
26 год	M3 304 нержавеющая сталь Самоконтрящиеся гайки	4	
27	Винты с полукруглой головкой M3 * 10 мм	20	
28 год	Винты с полукруглой головкой M2,5 * 10 мм	6	
29	Никелированные гайки M2.5	6	
30	Винты с полукруглой головкой M3 * 12 мм	6	
31 год	Винты с плоской головкой M3 * 10 мм	2	
32	M1,2 * 5 мм с круглой головкой Саморезы	10	
33	Держатель для батареек AA на 6 слотов с постоянным током Голова и линия росы 15 см	1	
34	Черный желтый Ручка 3 * 40 мм крест Отвертка	1	
35 год	20 см 2,54 3Pin FF перемычка	13	



36	20 см 2,54 4Pin FF перемычка	2	
----	------------------------------	---	---

3. Загрузите программное обеспечение и установите драйвер.

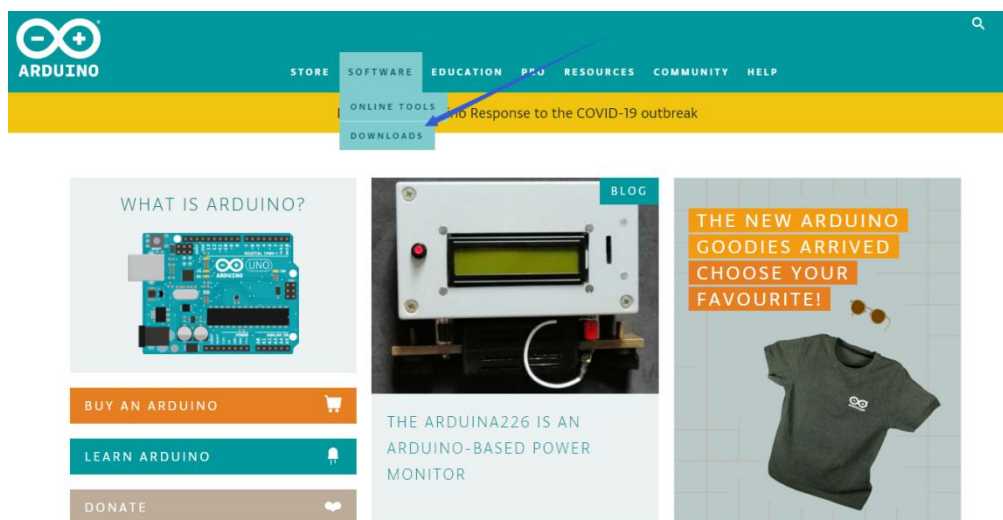
(1) Загрузить программное обеспечение

Когда мы получаем плату управления, нам в первую очередь нужно загрузить Arduino IDE и драйвер.

Вы можете скачать Arduino IDE с официального сайта:

<https://www.arduino.cc/>, и щелкните **ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ> ЗАГРУЗИТЬ**, как показано

ниже:



Вы можете выбрать последнюю версию ----- 1.8.13. Другой вариант - предыдущая версия.

В этом проекте мы используем версию 1.8.12.



Previous Releases

Download the previous version of the current release, the classic 1.0.x, or old beta releases.

DOWNLOAD OPTIONS

Previous Release (1.8.12)

Arduino 1.0.x

Arduino 1.5.x beta

Arduino 1.9.x beta

Previous Release (1.8.12)

Нажмите

для входа на новую страницу. Как показано ниже;

В **Установщик Windows** требует установки вручную. Тем не менее, **ZIP-файл Windows**

для установки без администратора, zip-файл версии Arduino 1.8.12, может быть

загружен и установлен напрямую.



Previous IDE Releases

ARDUINO 1.8.12

Arduino IDE that can be used with any Arduino board, including the Arduino Yún and Arduino DUE. Refer to the [Getting Started](#) page for Installation instructions.

[See the release notes.](#)

Windows Installer

Windows ZIP file for non admin install

Mac OS X 10.8 Mountain Lion or newer

Linux 32 bits

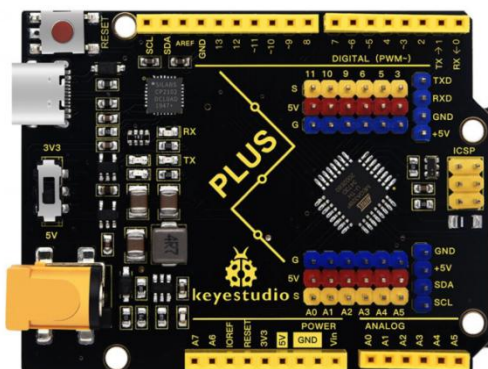
Linux 64 bits

Linux ARM 32

Linux ARM 64

[Source](#)

(2) Плата разработки Keyestudio PLUS

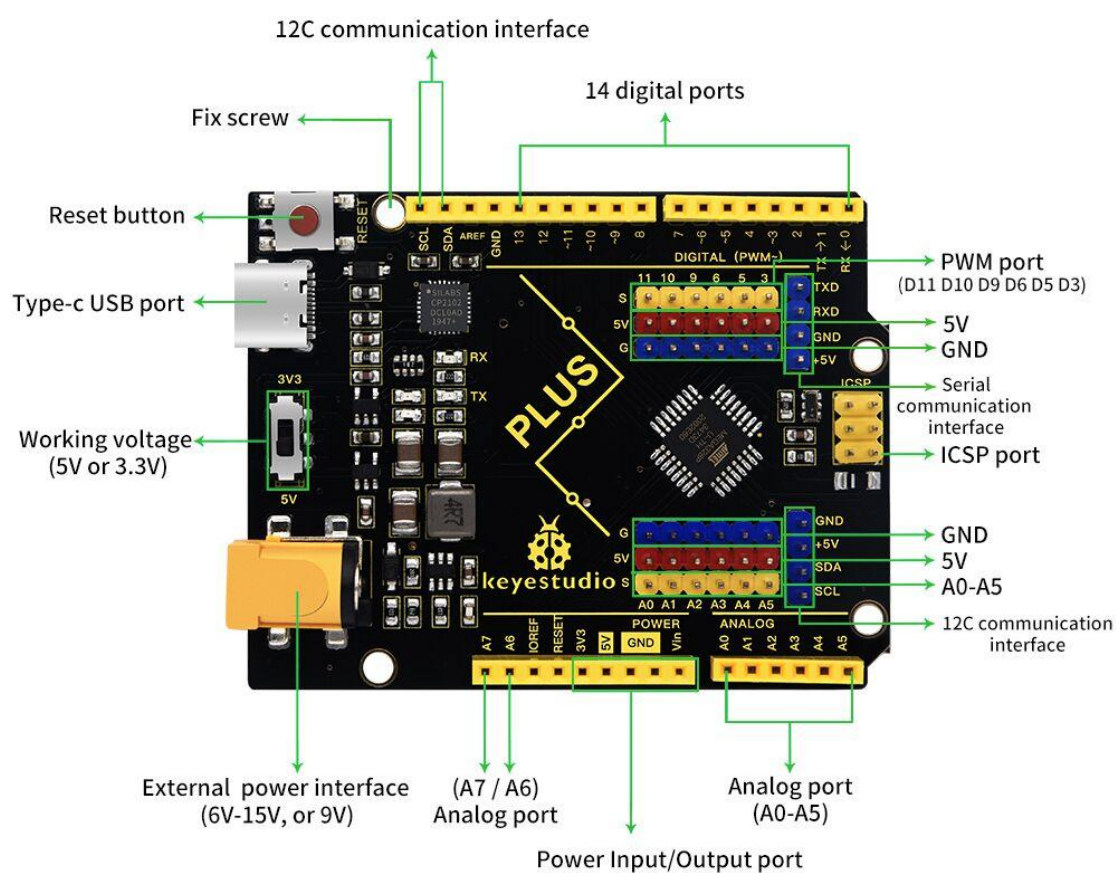


Теперь давайте познакомимся с платой разработки Keyestudio PLUS. Это ядро



весь комплект.

Плата управления Keyestudio PLUS полностью совместима с платой Arduino UNO R3. Его функции такие же, как у платы Arduino UNO R3. Более того, некоторые улучшения значительно усилили его функцию. В качестве альтернативы, это лучший выбор для изучения построения схем и проектирования кодов.



Интерфейс последовательной связи: D0 - RX, D1 - TX

Интерфейс ШИМ (широотно-импульсная модуляция): D3 D5 D6 D9 D10 D11

Интерфейс внешнего прерывания: D2 (прерывание 0) и D3 (прерывание 1)

Интерфейс связи SPI: D10 - SS, D11 - MOSI, D12 - MISO, D13 - SCK



Порт связи IIC: A4 - SDA, A5 - SCL

(3) Установка драйвера

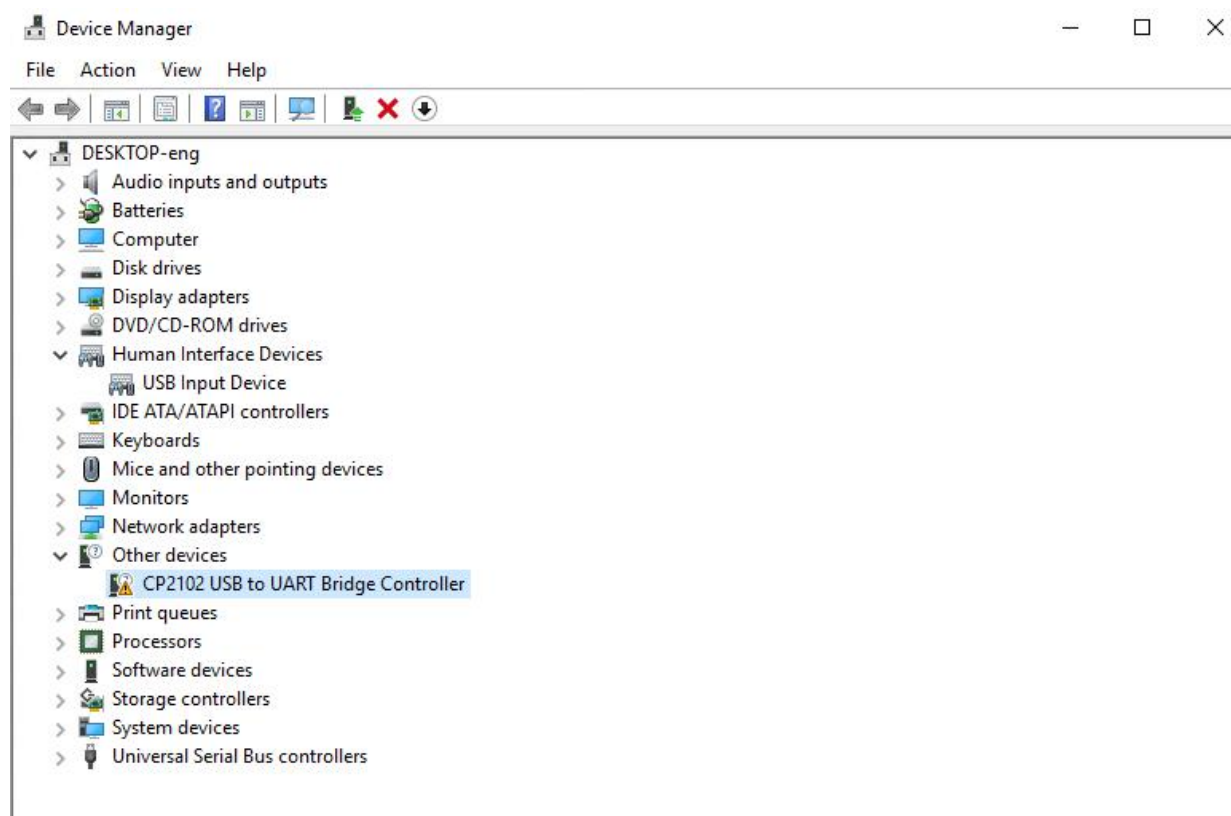
Установим драйвер Keyestudio PLUS Control Board. Чип USB-TTL на плате PLUS использует серийный чип CP2102. Программа драйвера этого чипа включена в версию Arduino 1.8 и выше, что удобно.

Когда вы подключаете USB-порт к компьютеру, можно установить драйвер CP2102.

Если драйвер устанавливается безуспешно, необходимо установить его вручную.

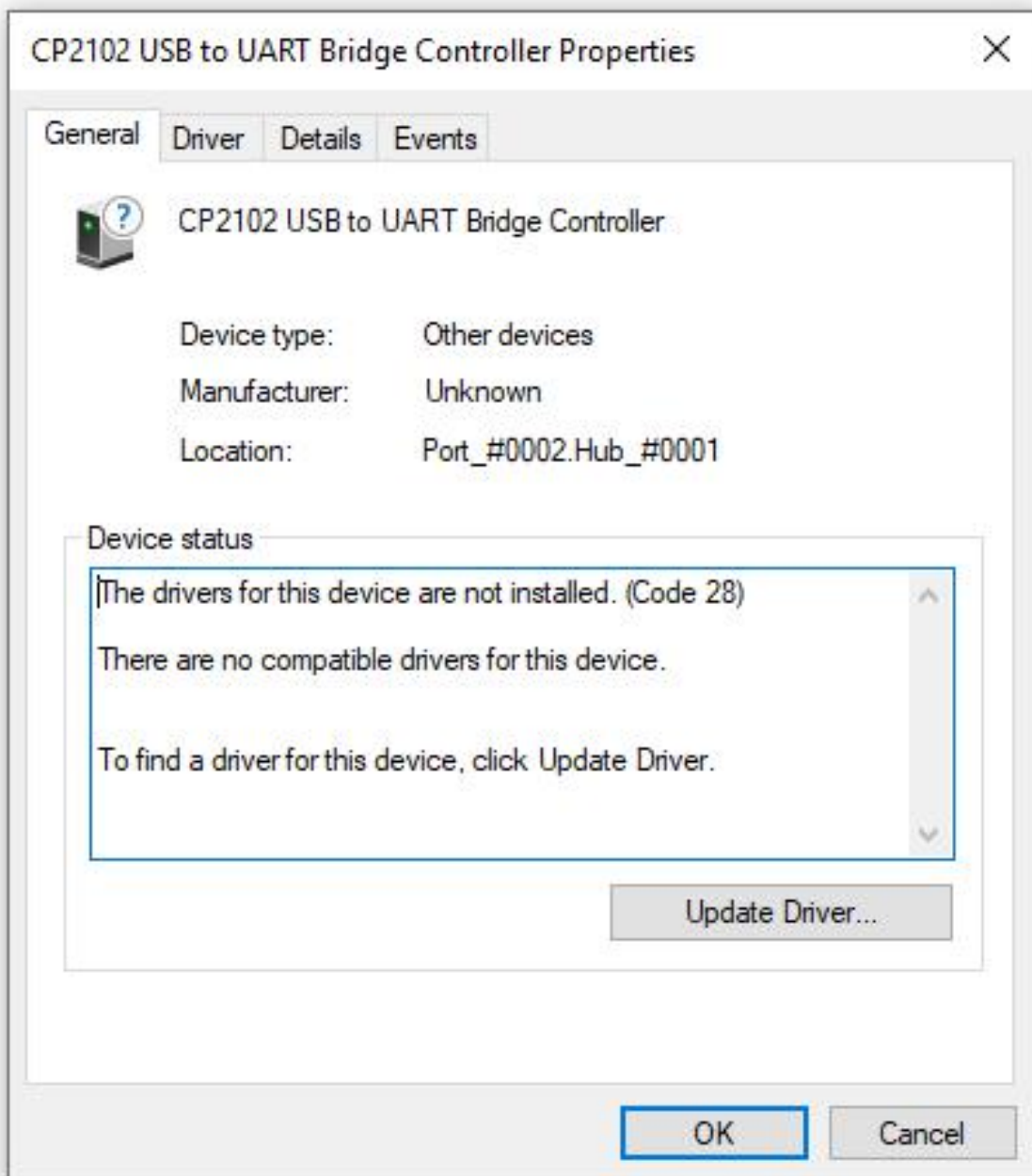
Откройте диспетчер устройств компьютера. Щелкните правой кнопкой мыши Компьютер ----- Свойства ----

Диспетчер устройств.

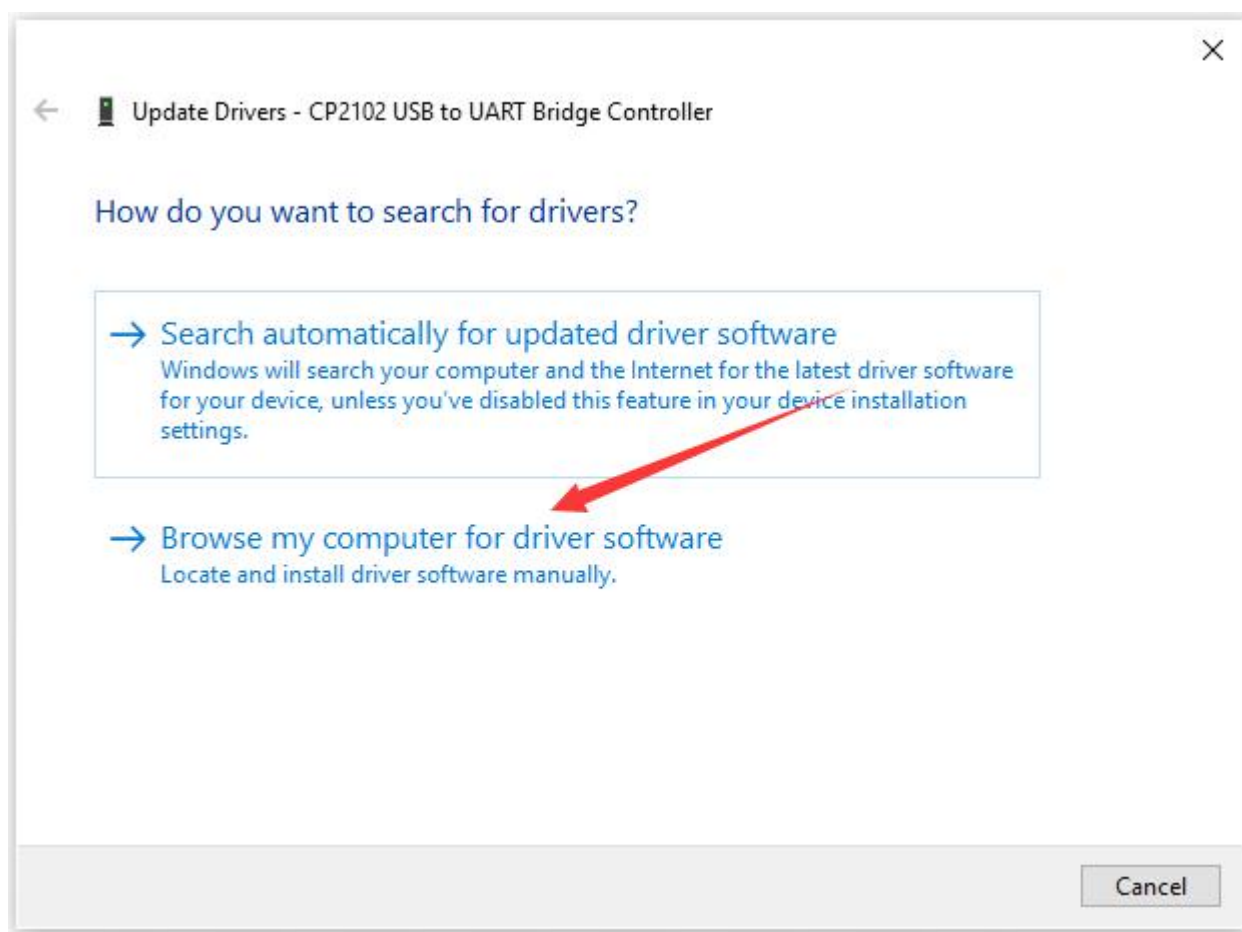


На странице есть желтый восклицательный знак, что означает сбой установки драйвера CP2102.

Действуйте следующим образом;




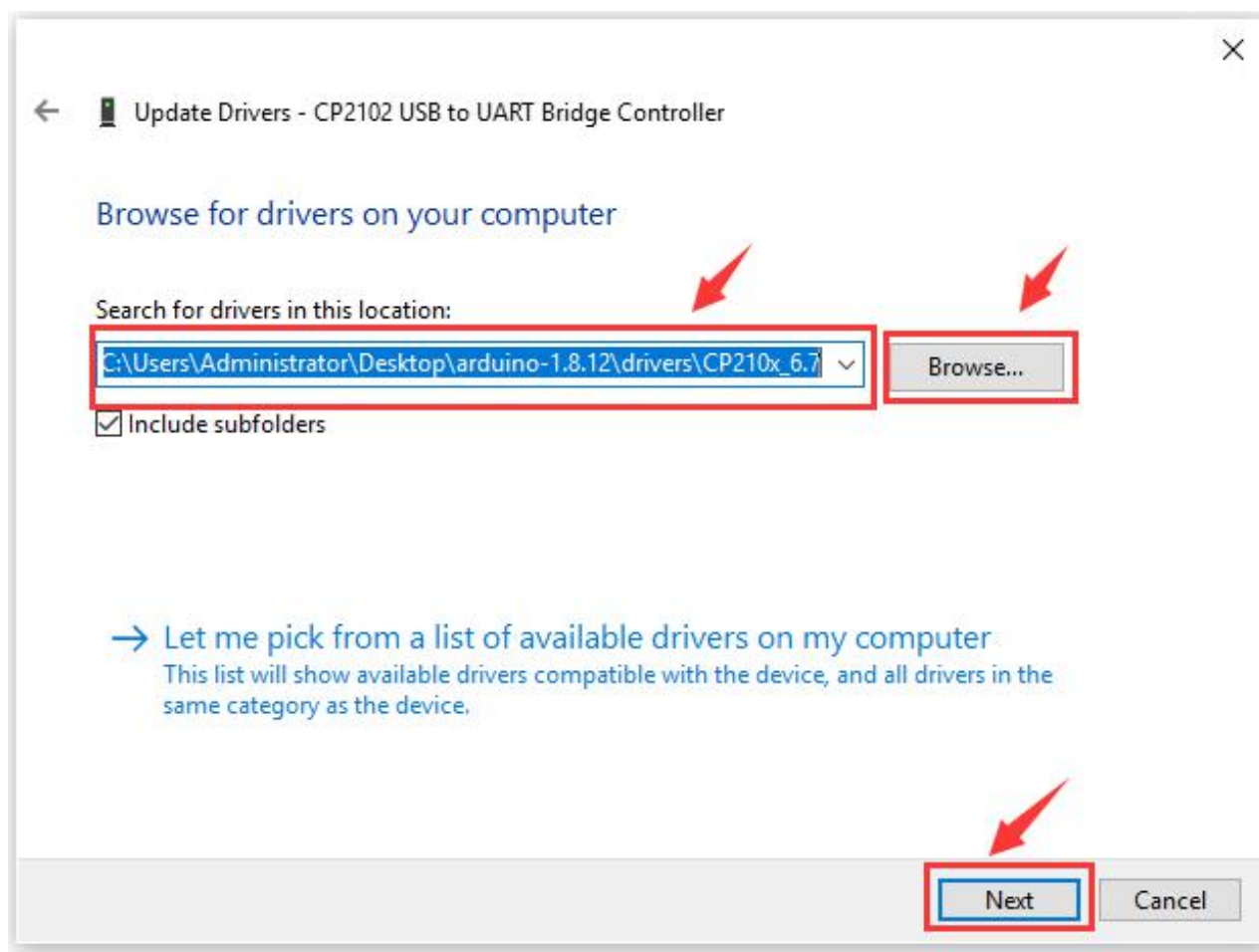
Нажмите «OK», чтобы перейти на следующую страницу, и нажмите «найти на моем компьютере обновленное программное обеспечение драйвера».



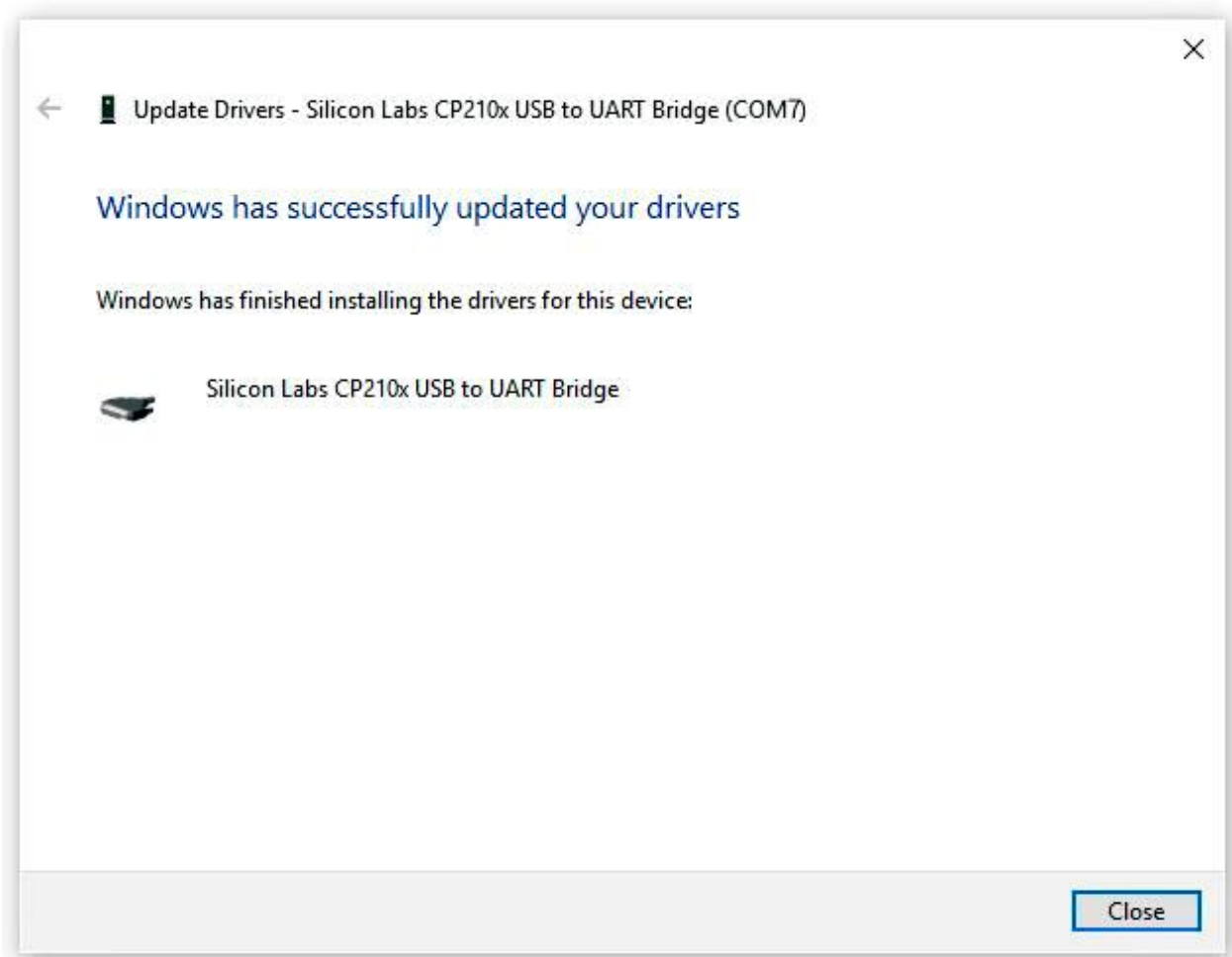
Нажмите «Обзор», затем найдите драйвер CP2102 и нажмите «Далее»,

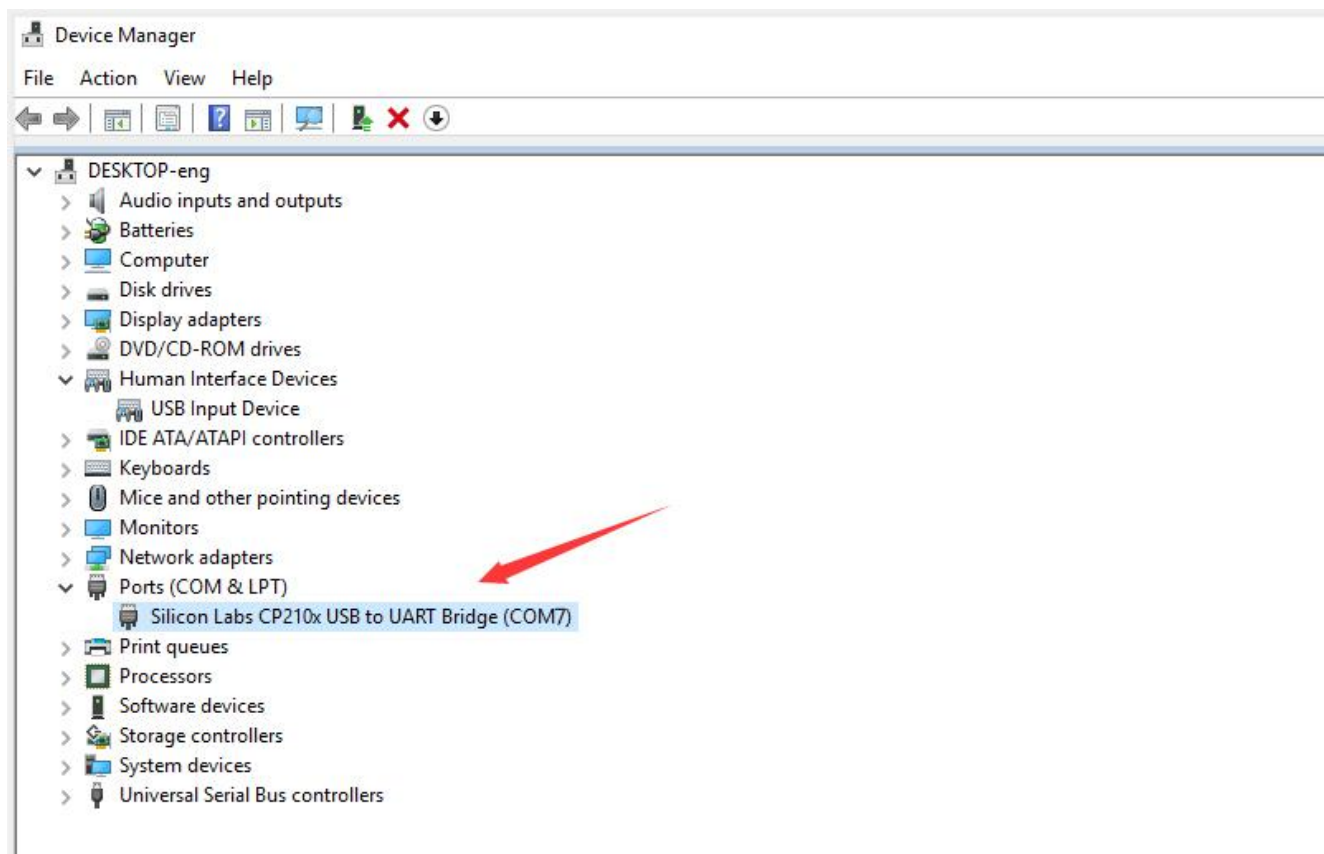
Есть [ВОДИТЕЛИ](#) папка в [Установленный пакет программного обеспечения Arduino](#)

( arduino-1.8.12) , откройте папку с драйверами и проверьте драйвер [CP210X серии](#)
[чипсы](#).



При открытии диспетчера устройств желтый восклицательный знак исчезает. Драйвер CP2102 успешно установлен.

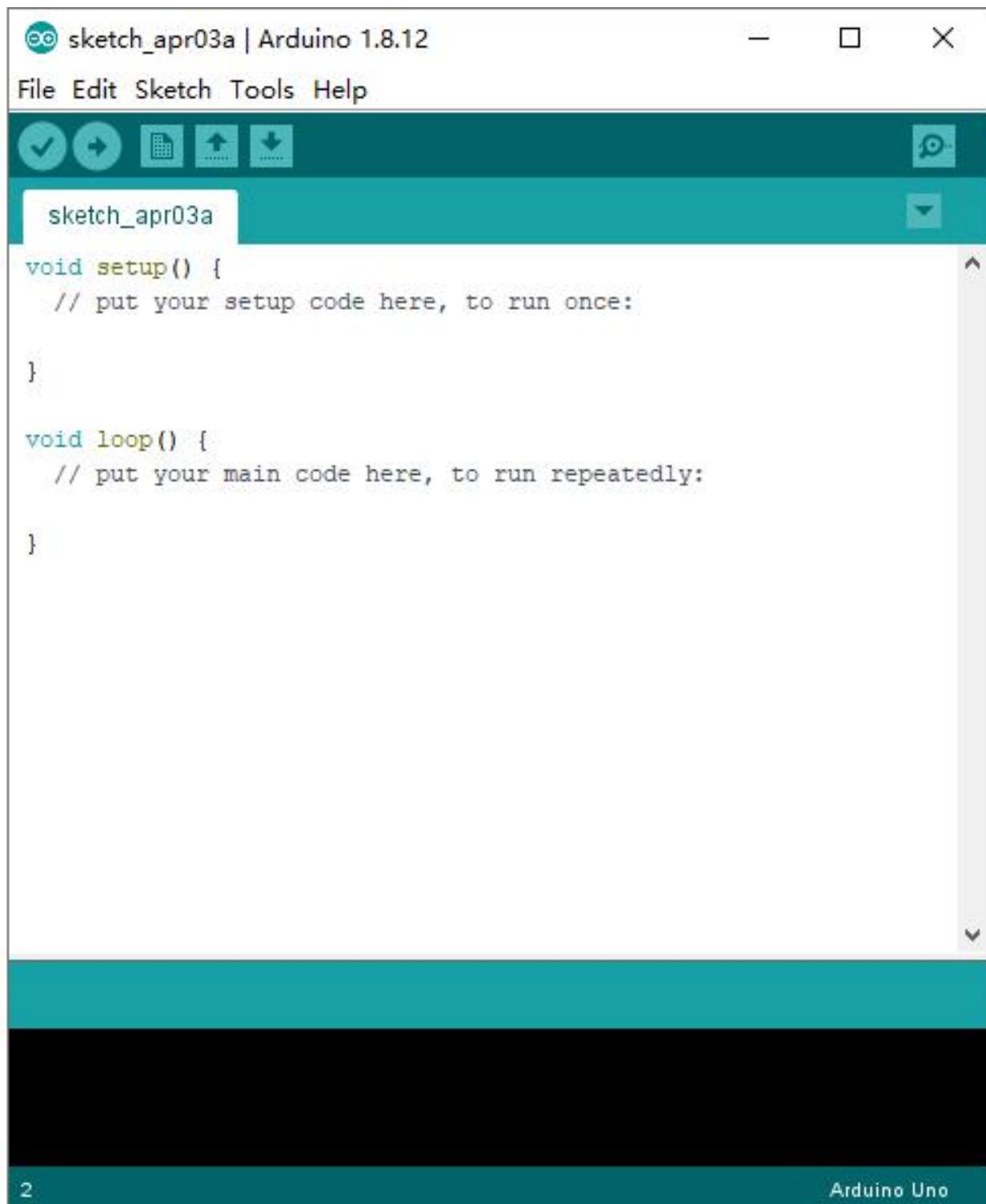




(4) Настройка Arduino IDE



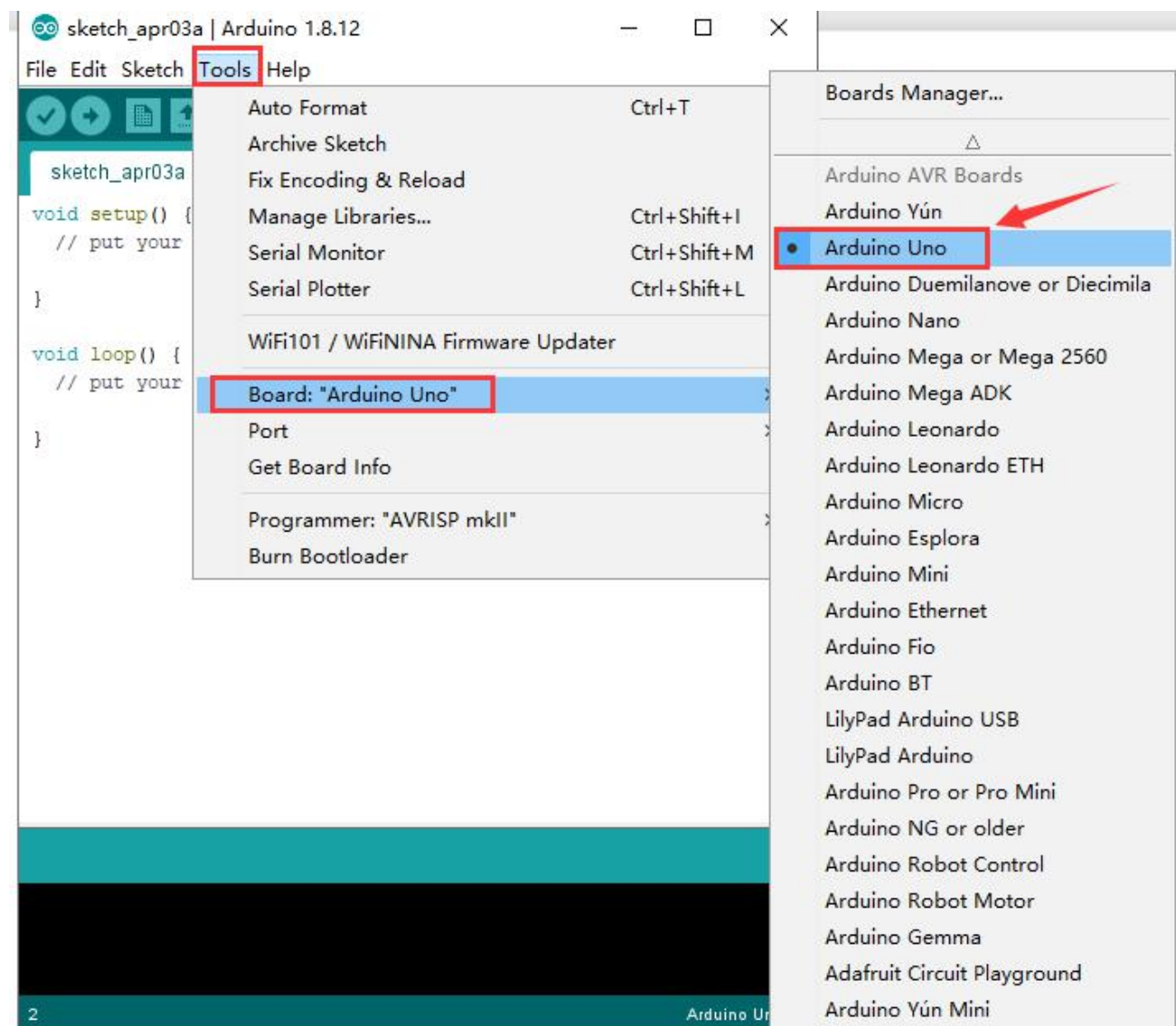
Нажмите значок , откройте Arduino IDE.



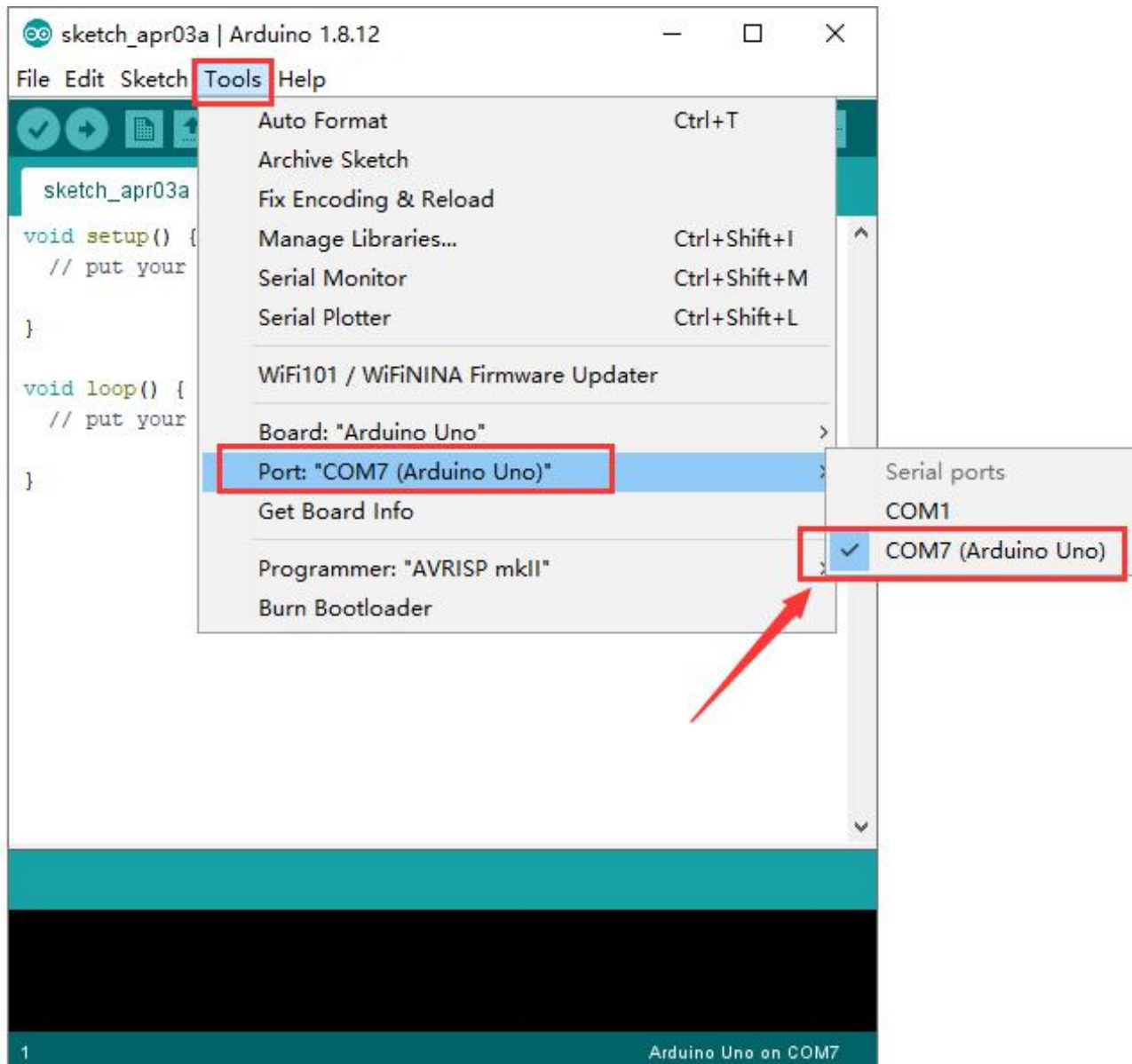
При загрузке скетча на плату вы должны выбрать правильное имя платы Arduino, которое соответствует плате, подключенной к вашему

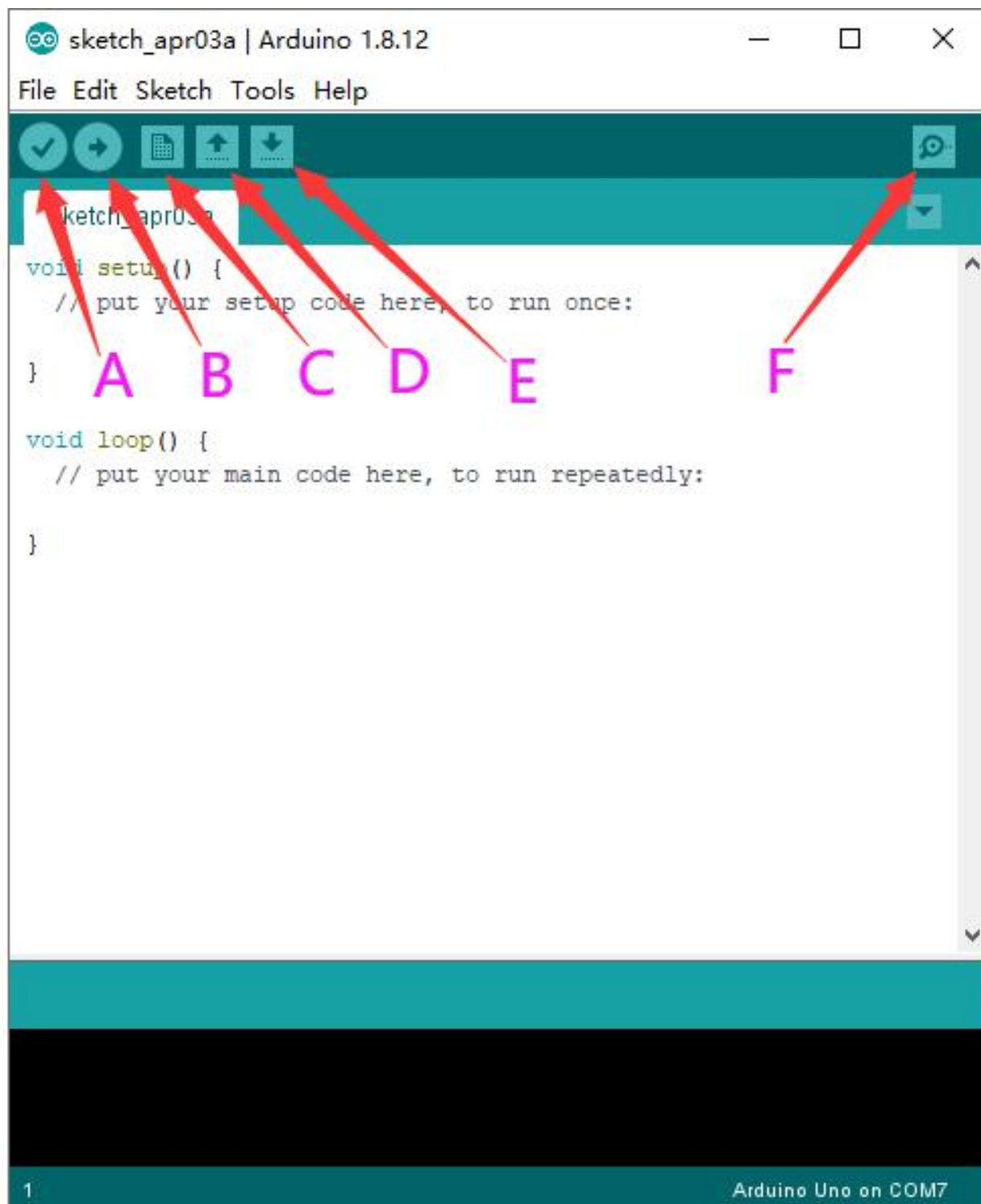


компьютер. Как показано ниже;



Затем выберите правильный COM-порт (вы увидите соответствующий COM-порт после успешной установки драйвера)





A- Используется для проверки наличия ошибок компиляции. B-
Используется для загрузки эскиза на плату Arduino.



C- Используется для создания окна быстрого доступа к новому эскизу. D-

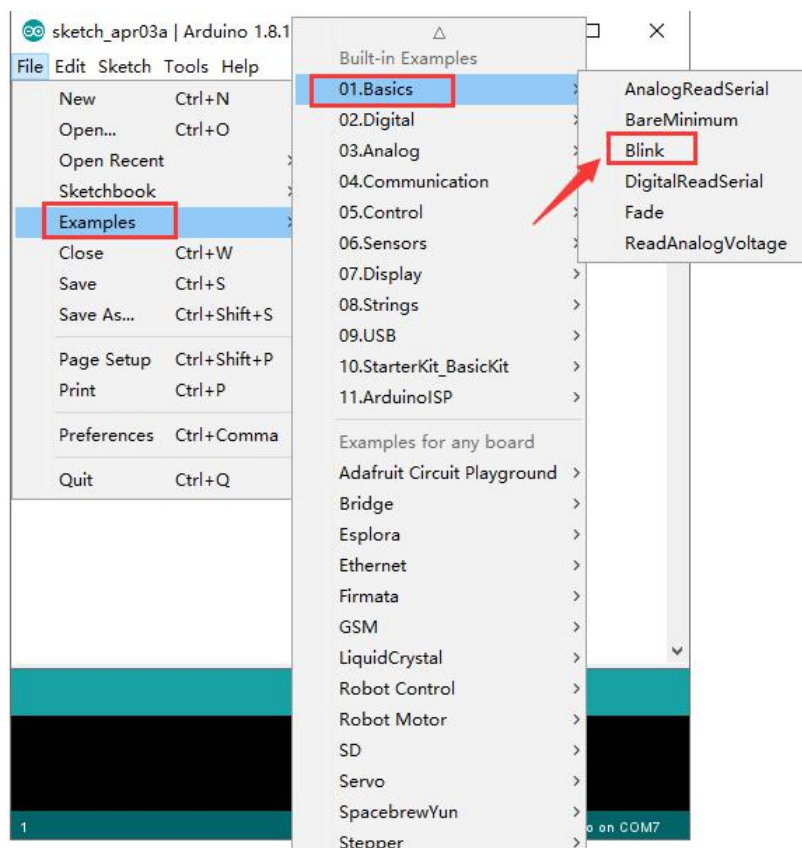
Используется для непосредственного открытия примерного скетча. E-

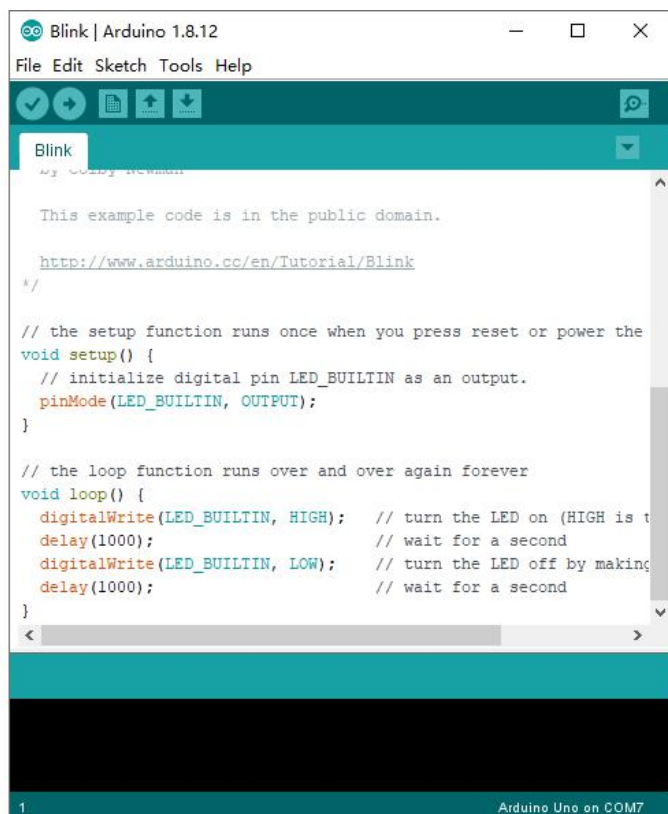
Используется для сохранения эскиза.

F- Используется для отправки последовательных данных, полученных с платы, на последовательный монитор.

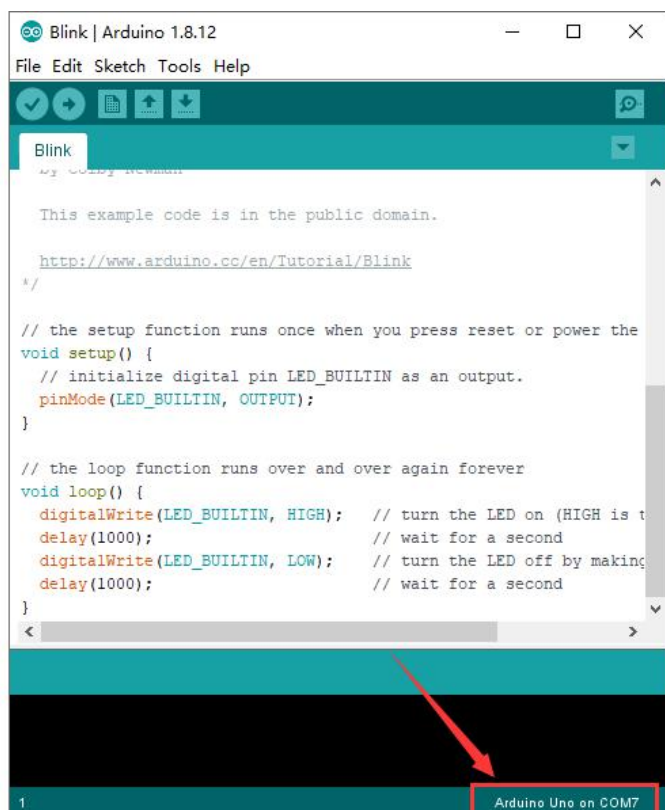
(5) Запустите свою первую программу

Откройте файл, чтобы выбрать **Пример**, и нажмите **ОСНОВНОЙ> МИГАНИЕ**, как показано ниже:





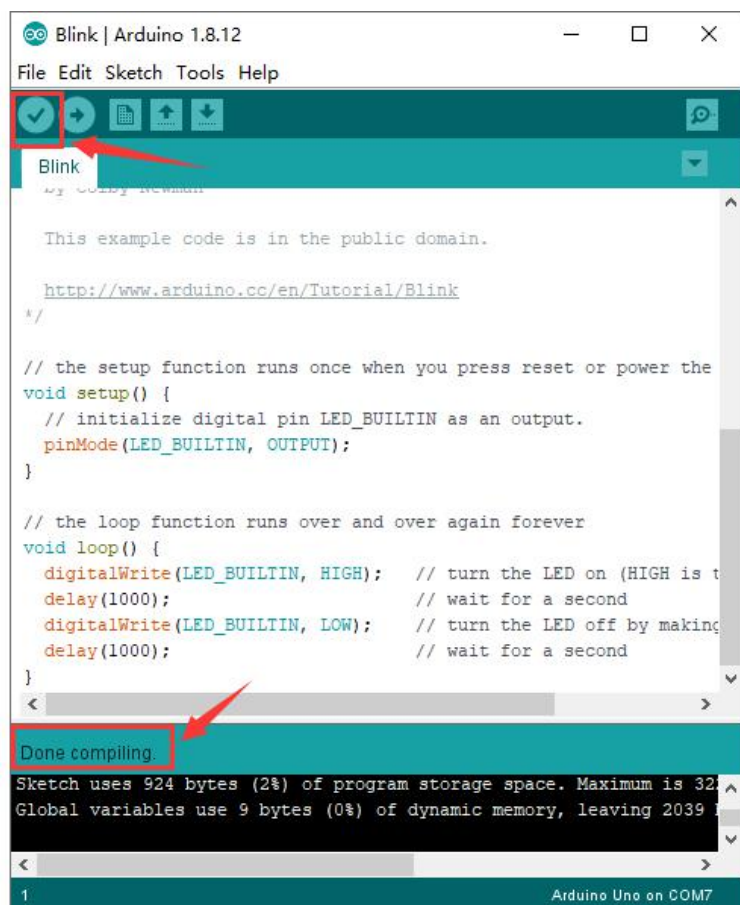
Установите правильный **COMport**, а соответствующая плата и COM-порт показаны в правом нижнем углу IDE.





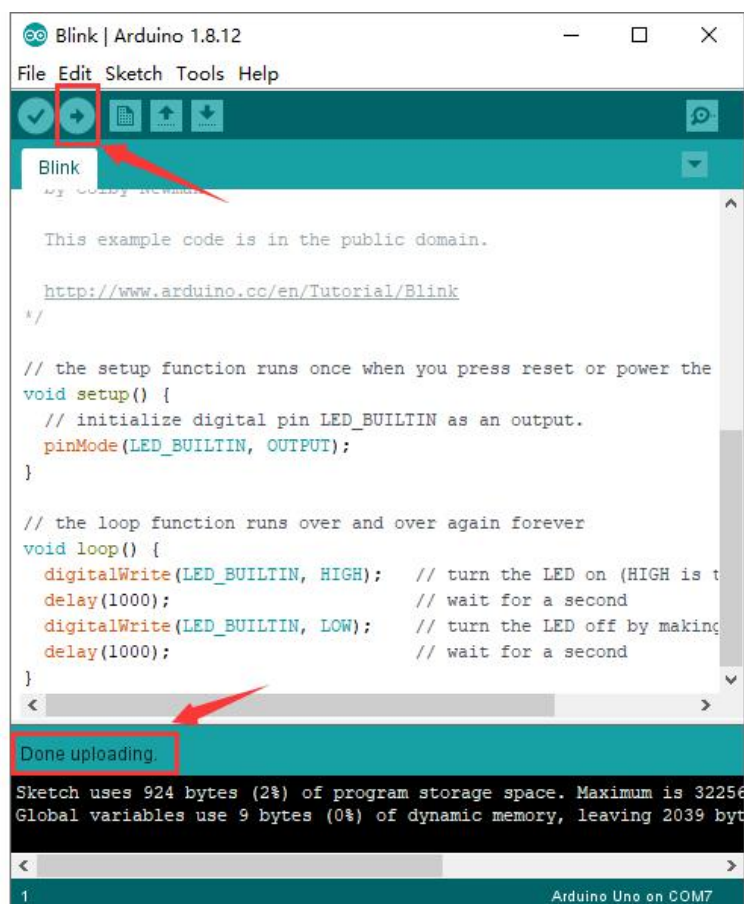
Нажмите

начать компилировать программу и проверить ошибки.



Нажмите

загрузить программу



После успешной загрузки программы на плате мигает светодиод.

Поздравляю, вы закончили первую программу.

4. Как добавить библиотеку?

Что такое библиотеки?

Библиотеки - это набор кода, который упрощает управление датчиками, дисплеями, модулями и т. Д.

В Интернете для загрузки доступны сотни дополнительных библиотек.

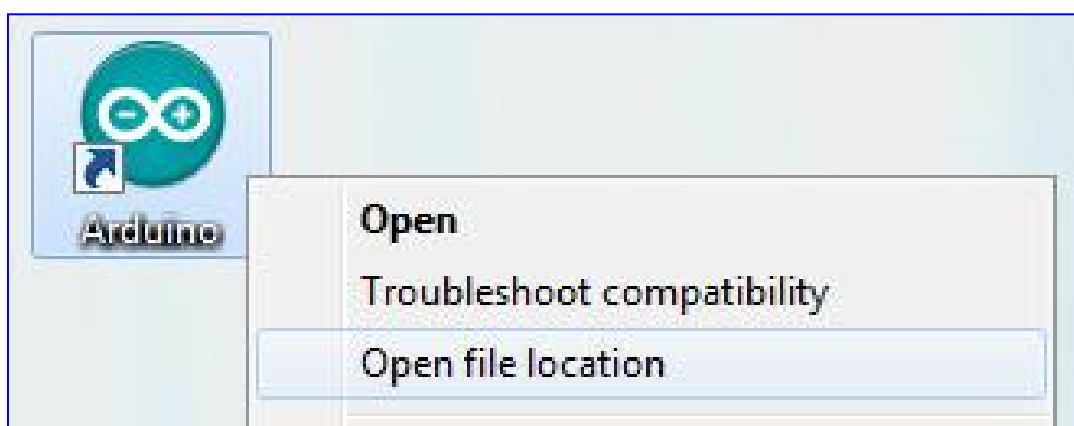
Встроенные библиотеки и некоторые из этих дополнительных библиотек перечислены в справочнике.



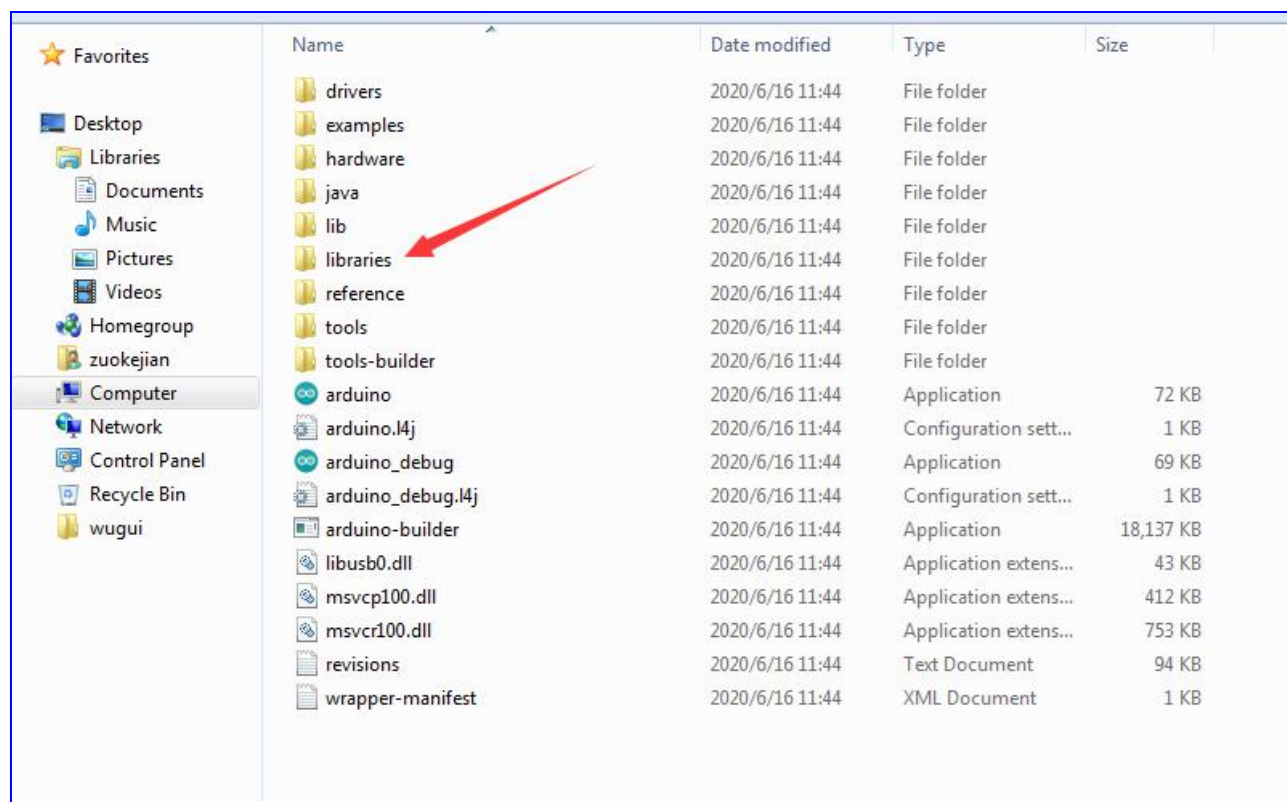
Здесь мы познакомим вас с самым простым способом добавления библиотек.

Шаг 1. После правильной загрузки IDE Arduino вы можете щелкнуть правой кнопкой мыши значок Arduino IDE.

Найдите опцию «Открыть расположение файла», как показано ниже:



Шаг 2: Войдите в папку библиотек Arduino, как показано ниже;



Шаг 3 : Затем найдите «Библиотеки» комплекта умного дома (видно по ссылке:

<https://fs.keyestudio.com/KS0085>), как показано ниже:



> Starter kit > KS0085 Smart Home Kit for Arduino








Overview

[Hide](#)

Click here to describe this folder and turn it into a Space

[Show examples](#)[Create new file](#)

Name ↑	Modified ▾	Members ▾	⋮ ▾
 1. About keyestudio	--	Only you	...
 2. Tutorial for Arduino	--	Only you	...
 3. APP	--	Only you	...
 4. Tutorial for Mixly	--	Only you	...
 5. Installation guide	--	Only you	...



> 2. Tutorial for Arduino



Overview

[Hide](#)

Click here to describe this folder and turn it into a Space

[Show examples](#)[Create new file](#)

Name ↑	Modified ▾	Members ▾	⋮ ▾
1. Arduino Software	--	Only you	...
2. Getting Started With Arduino	--	Only you	...
3. Tutorial	--	Only you	...
4. Arduino code	--	Only you	...
5. Arduino library files	--	Only you	...

> 2. Tutorial for Arduino > 5. Arduino library files

Overview

Click here to describe this folder and turn it into a Space

[Show examples](#)[Create new file](#)

✓	Name ↑	Modified ▾	Members ▾
✓	LiquidCrystal_I2C	--	Only Share ▾
✓	Servo	--	Only Share ▾



Вам просто нужно скопировать и вставить

в библиотеки

папка Arduino IDE.

Затем библиотеки домашнего смарта успешно устанавливаются, как показано

ниже:

NAME	DATE MODIFIED	TYPE
Adafruit_Circuit_Playground	2020/2/13 10:32	文件夹
Bridge	2020/2/13 10:32	文件夹
Esplora	2020/2/13 10:32	文件夹
Ethernet	2020/2/13 10:32	文件夹
Firmata	2020/2/13 10:32	文件夹
GSM	2020/2/13 10:32	文件夹
IRremote	2020/8/18 14:15	文件夹
Keyboard	2020/2/13 10:32	文件夹
LiquidCrystal	2020/2/13 10:32	文件夹
LiquidCrystal_I2C	2020/8/26 9:38	文件夹
Mouse	2020/2/13 10:32	文件夹
Robot_Control	2020/2/13 10:32	文件夹
Robot_Motor	2020/2/13 10:32	文件夹
RobotIRremote	2020/2/13 10:32	文件夹
SD	2020/2/13 10:32	文件夹
Servo	2020/2/13 10:32	文件夹
SpacebrewYun	2020/2/13 10:32	文件夹
SR04	2020/8/17 15:51	文件夹
Stepper	2020/2/13 10:32	文件夹
Temboo	2020/2/13 10:32	文件夹
TFT	2020/2/13 10:32	文件夹
WiFi	2020/2/13 10:32	文件夹
.keep	2020/2/13 10:32	KEEP 文件



5. Проекты



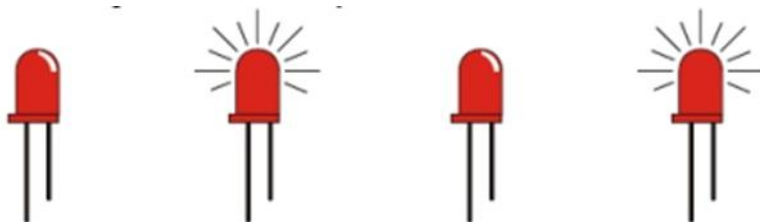
Хорошо, перейдем к нашим проектам. В этом комплекте 14 датчиков и модулей.

Мы познакомим вас с умным домом с помощью простого датчика. Однако, если вы профессионально работаете с Arduino. Вы можете пропустить эти шаги и собрать комплект умного дома напрямую ([в папке есть сборочное видео](#))

Примечание. В этом случае интерфейс каждого датчика / модуля, отмеченный (G, -, GND), указывает отрицательный полюс, G подключен к G, - или GND экрана датчика или платы управления; «V» - положительный полюс и связан с V, VCC или 5V.



Проект 1: мигание светодиода



1. Описание

Мы установили драйвер платы разработки Keyestudio V4.0. В

этом уроке мы проведем эксперимент, чтобы светодиод мигал.

Подключим GND и VCC к питанию. Светодиод будет гореть, когда на конце сигнала S будет высокий уровень, и наоборот, светодиод погаснет, когда на конце сигнала S будет низкий уровень.

Кроме того, различная частота мигания может быть представлена путем настройки времени задержки.

2. Технические характеристики

Интерфейс управления: цифровой порт

Рабочее напряжение: 3,3-5 В постоянного

тока Шаг контактов: 2,54 мм

Цвет светодиодного дисплея:

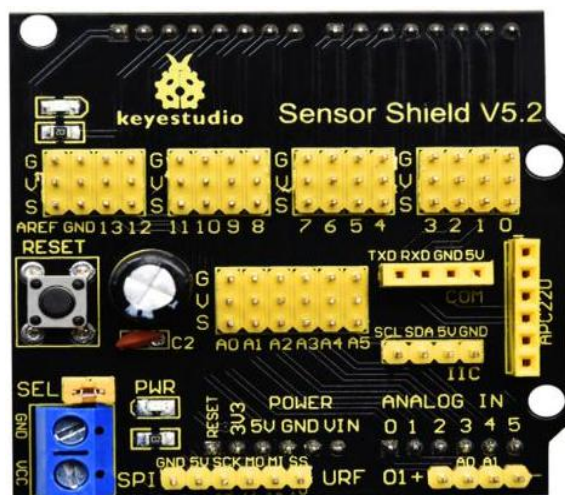
белый Цвет дисплея: белый



3. Что вы Нужно

PLUS Control	Датчик	Белый светодиод	USB	3-контактный FF Dupont
Доска * 1	Щит * 1	Модуль * 1	Кабель * 1	Кабель * 1
				

4. Сенсорный экран



Обычно мы комбинируем плату управления Arduino с большим количеством датчиков и модулей.

Однако количество контактов и портов на плате управления ограничено.

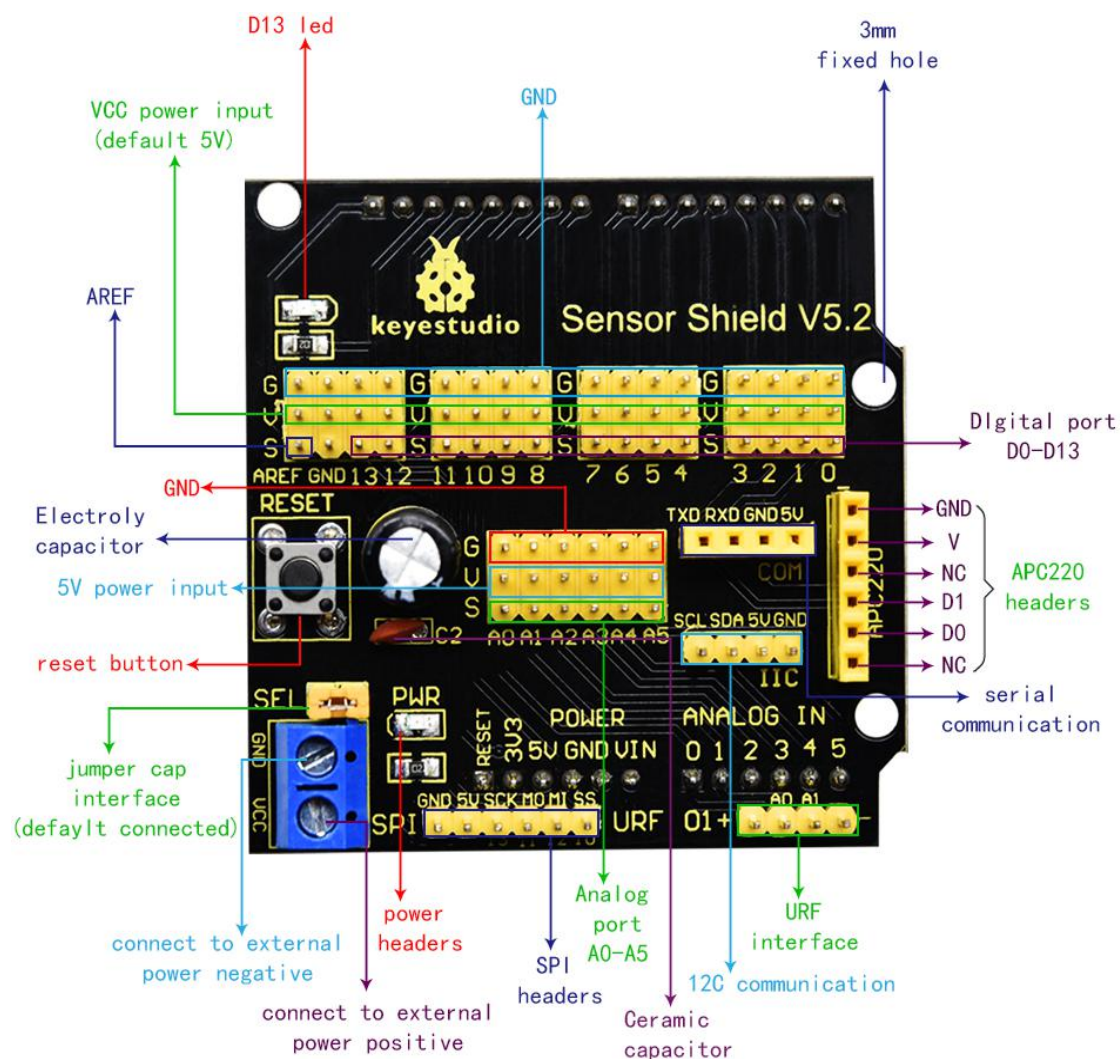
Чтобы справиться с этим недостатком, нам просто нужно установить плату датчика V5 на плату управления Keyestudio PLUS.

Этот экран V5 может быть непосредственно присоединен к датчикам с помощью 3-контактных разъемов, а также может быть расширен до часто используемых портов связи, таких как последовательная связь, связь IIC и порты связи SPI. Более того,



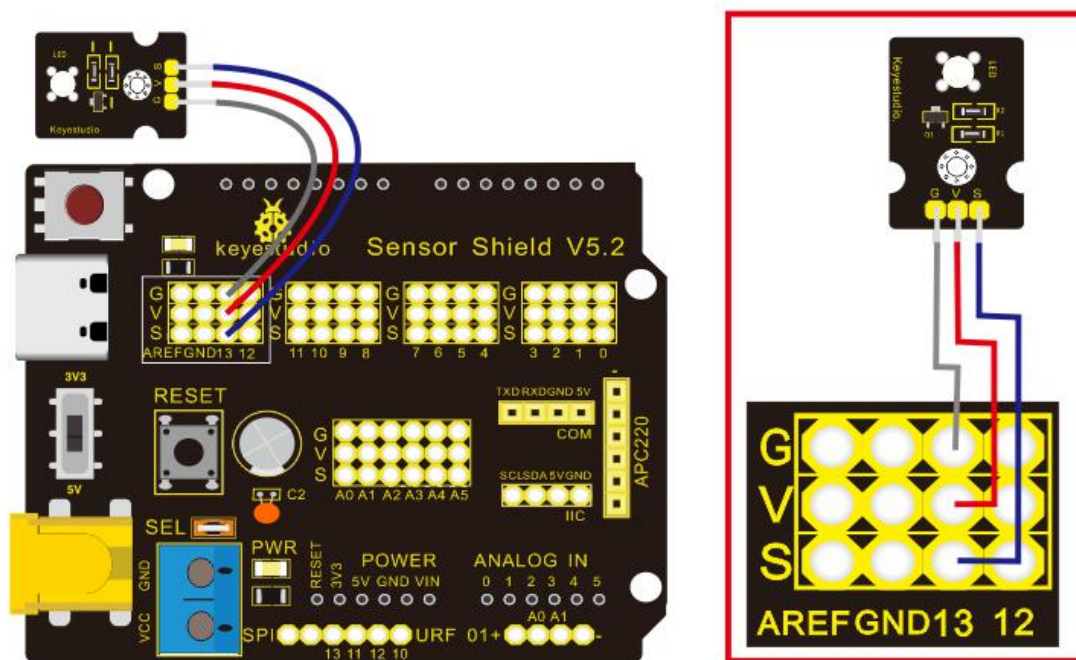
щиток поставляется с кнопкой сброса и 2 сигнальными лампами.

Описание контактов



5. Схема подключения

Свяжите светодиодный модуль с D13 экрана.



Примечание: контакты G, V и S модуля белого светодиода связаны с G, V и 13 на плате V5.

6. Тестовый код

/ *

Комплект умного дома Keyestudio для Arduino

Project 1

Мигать

<http://www.keyestudio.com>

* /

```
void setup () {
```

```
    // инициализируем цифровой вывод 13 как
```

```
    выход. pinMode (13, ВЫХОД);
```



```
}  
  
// функция цикла запускается снова и снова навсегда  
  
void loop () {  
  
    digitalWrite (13, ВЫСОКИЙ);      // включаем светодиод (HIGH - уровень  
  
    задержка (1000);                  напряжения) // ждем секунду  
  
    digitalWrite (13, LOW);           // выключаем светодиод, понижая  
  
    задержка (1000);                  напряжение // ждем секунду  
  
} // *****  
  
*****
```

7. Результат теста:

После загрузки кода белый светодиод будет попеременно мигать в течение 1000 мс.

8. Расшифровка кода

Код выглядит длинным и беспорядочным, но по большей части это комментарии. Грамматика Arduino основана на C.

Комментарии обычно имеют две формы выражения: / *

* /: подходит для длинных комментариев абзаца /:

подходит для однострочных комментариев

Код содержит много важной информации, такой как автор, подписанный договор и т. Д.

Начинающий должен выработать хорошую привычку просматривать код.



Комментарии, составляющие большую часть всего кода, содержат важную информацию и действительно помогают быстро понять тестовый код.

```
// функция настройки запускается один раз, когда вы нажимаете кнопку сброса или включаете плату
```

```
void setup () {
```

```
    // инициализируем цифровой вывод 13 как
```

```
    выход. pinMode (13, ВЫХОД);
```

```
}
```

Согласно комментариям, мы обнаружим, что автор определяет режим вывода D13 как цифровой выход в функции setup ().

Setup () - это основная функция Arduino, которая выполняется один раз при запуске программы.

```
// функция цикла запускается снова и снова навсегда
```

```
void loop () {
```

```
    digitalWrite (13, ВЫСОКИЙ);    // включаем светодиод (HIGH - уровень
```

```
    задержка (1000);                напряжения) // ждем секунду
```

```
    digitalWrite (13, LOW);         // выключаем светодиод, понижая
```

```
    задержка (1000);                напряжение // ждем секунду
```

```
}
```




Loop () - необходимая функция Arduino, она может запускаться и за цикливаться все время после того, как

«setup ()» выполнится один раз.

В функции loop () автор использует

`digitalWrite (13, ВЫСОКИЙ); // включаем светодиод (HIGH - уровень напряжения)`

digitalWrite (): установите выходное напряжение вывода на высокий или низкий уровень. Делаем вывод D13

высоким уровнем, после чего загорается светодиод.

`задержка (1000); // ждем секунду`

Функция задержки используется для времени задержки, 1000 мс - 1 с, единица - мс.

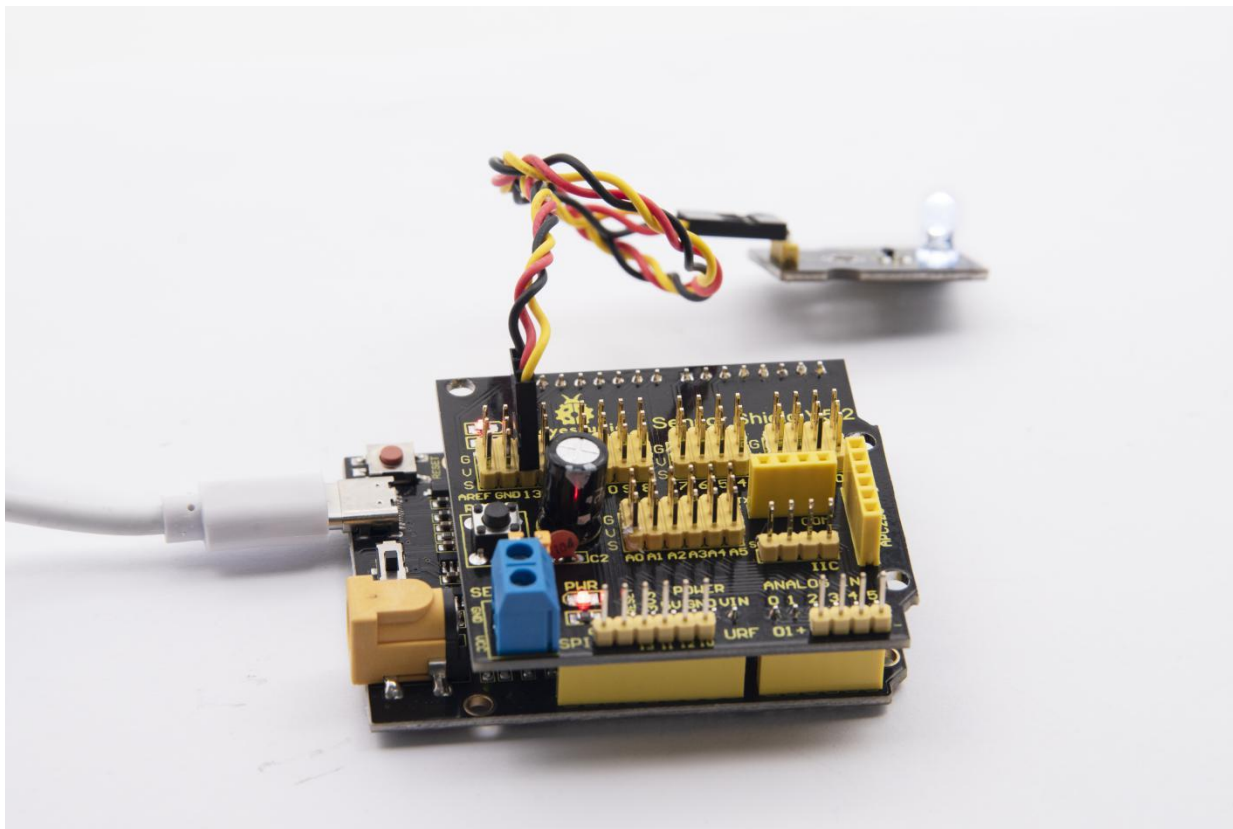
`digitalWrite (13, LOW); // выключаем светодиод, понижая напряжение`

Аналогично делаем вывод D13 низким уровнем, светодиод погаснет.

`задержка (1000); // ждем секунду`

Задержка на 1 с, светодиод горит - держится 1 с - светодиод не горит - остается на 1 с, итерация процесса. Светодиод мигает с интервалом в 1 секунду.

Что делать, если вы хотите, чтобы светодиод быстро мигал? Вам нужно только изменить значение блока задержки. Уменьшение значения задержки означает, что время ожидания короче, то есть быстрое мигание. И наоборот, вы можете заставить светодиод мигать медленно.



Проект 2: Дышащий свет



Breathing light

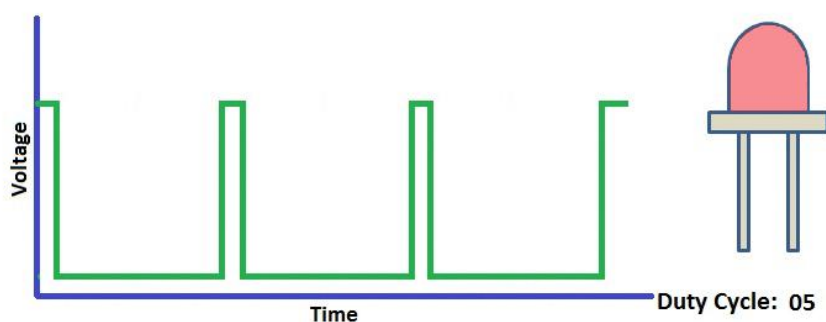
1. Описание



В предыдущем уроке мы управляли включением и выключением светодиода и заставляли его мигать.

В этом проекте мы будем управлять яркостью светодиода с помощью ШИМ для имитации эффекта дыхания. Точно так же вы можете изменить длину шага и время задержки в коде, чтобы продемонстрировать другой эффект дыхания.

ШИМ - это средство управления аналоговым выходом с помощью цифровых средств. Цифровое управление используется для генерации прямоугольных сигналов с различными рабочими циклами (сигнал, который постоянно переключается между высоким и низким уровнями) для управления аналоговым выходом. Как правило, входное напряжение порта составляет 0 В и 5 В. Что делать, если требуется 3В? Или что, если переключиться между 1 В, 3 В и 3,5 В? Мы не можем постоянно менять резистор. Для этой ситуации нам нужно управление с помощью ШИМ.



Для выхода напряжения цифрового порта Arduino есть только LOW и HIGH, которые соответствуют выходному напряжению 0 В и 5 В. Вы можете определить LOW как 0 и HIGH как 1, и позволить Arduino выводить пятьсот сигналов 0 или 1 в течение 1 секунды.

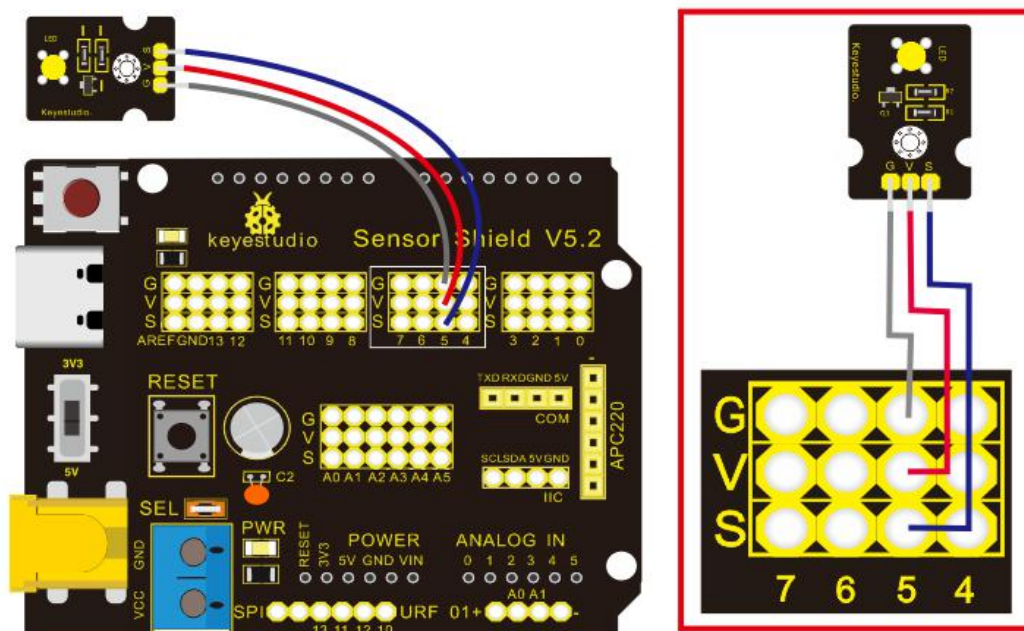


Если вывести пятьсот 1, то это 5В; если все это 1, то это 0 В. Если вывести 010101010101 таким образом, то выходной порт будет 2,5 В, что похоже на показ фильма. Фильм, который мы смотрим, не является полностью непрерывным. Фактически он выводит 25 изображений в секунду. В этом случае человек не может сказать этого, как и ШИМ. Если требуется другое напряжение, необходимо контролировать соотношение 0 и 1. Чем больше 0,1 сигналов выход за единицу времени, тем точнее контроль.

2. Что вам нужно

PLUS Control	Датчик	Желтый светодиод	USB-кабель * 1	3-контактный FF Dupont
Доска * 1	Щит * 1	Модуль * 1		Кабель * 1
				

3. Схема подключения



Примечание: на экране датчика контакты G, V и S желтого светодиодного модуля соединены с G, V и 5.

4. Тестовый код

/ *

Комплект умного дома Keyestudio для Arduino

Project 2

ШИМ

<http://www.keyestudio.com>

* /

```
int ledPin = 5; // Определяем вывод светодиода в
```

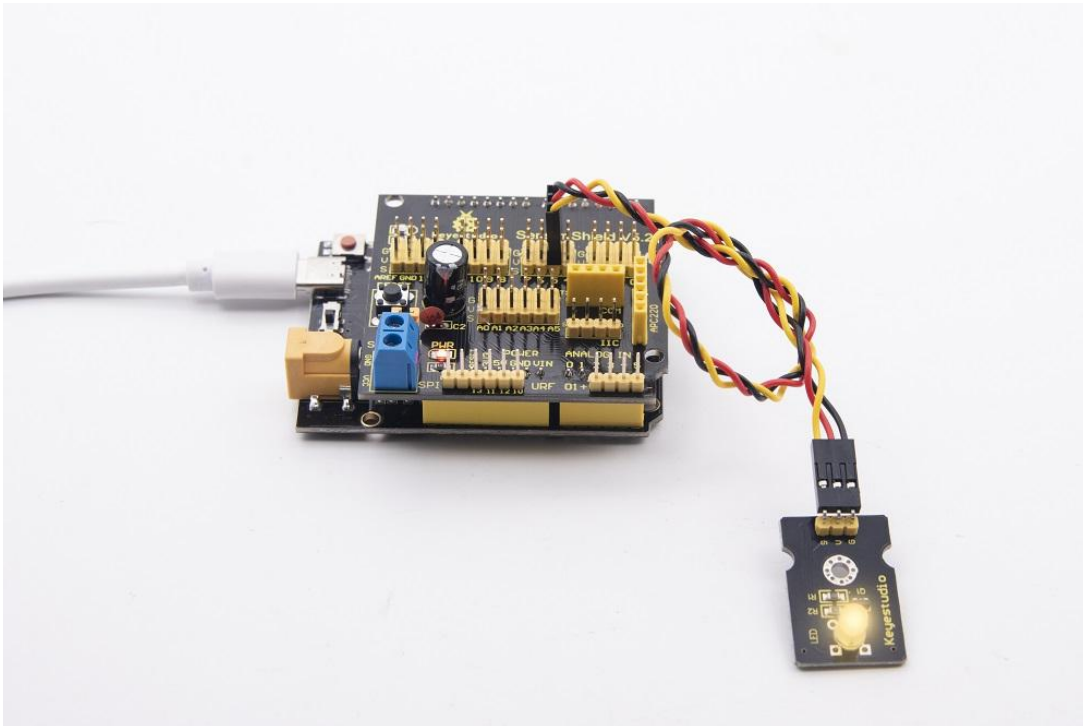
```
D5 void setup () {
```

```
    pinMode (ledPin, ВЫХОД); // инициализируем ledpin как выход.
```



```
}  
  
void loop () {  
  
  for (int value = 0; value <255; value = value + 1) {  
  
    analogWrite (ledPin, значение); // Светодиоды загораются постепенно  
  
    delay (5); // задержка 5 мс  
  
  }  
  
  для (целое значение = 255; значение > 0; значение = значение-1)  
  
  {analogWrite (ledPin, значение); // светодиод постепенно гаснет delay  
  
  (5); // задержка 5 мс  
  
  }  
  
  }  
  
  // *****
```

Светодиод плавно меняет свою яркость от темной к яркой и обратно к темной, продолжая это делать, что похоже на вдох и выдох легких.



Анализ кода

Когда нам нужно повторить некоторые операторы, мы должны использовать оператор «for» в формате оператора For следующим образом:

```
    ①      ② condition is true      ④  
for (cycle initialization; cycle condition; cycle adjustment statement) {  
    ③ loop body statement; ←  
}
```

«Для» циклической последовательности:

1 тур: 1 → 2 → 3 → 4

Раунд 2: 2 → 3 → 4...



Пока номер 2 не установлен, цикл for завершается.

Узнав этот порядок, вернитесь к коду:

```
for (int value = 0; value <255; value = value + 1) {  
  
    ...  
  
}  
  
for (int value = 255; value > 0; value = value-1) {  
  
    ...  
  
}
```

Два оператора «for» увеличивают значение от 0 до 255, затем уменьшают с 255 до 0, затем увеличивают до 255, бесконечный цикл.

В операторе for появилась новая функция ----- analogWrite ()

Мы знаем, что цифровой порт имеет только два состояния: 0 и 1. Итак, как передать аналоговое значение в цифровое значение? Здесь нам нужна эта функция, обратите внимание на плату Arduino, и вы найдете 6 контактов со знаком «~». Они отличаются от других контактов и могут выводить сигналы ШИМ.

Формат функции следующий:

`analogWrite (контакт, значение)`

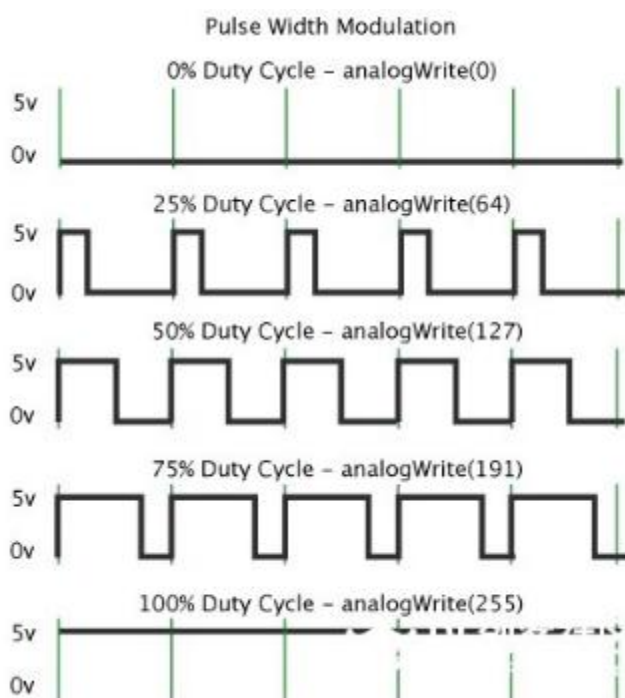
analogWrite () используется для записи аналогового значения от 0 до 255 для порта ШИМ, поэтому



значение находится в диапазоне 0 ~ 255, обратите внимание, что вы пишете только цифровые контакты с функцией PWM, такие как контакты 3, 5, 6, 9, 10, 11.

ШИМ - это технология получения аналоговой величины цифровым методом. Цифровое управление формирует прямоугольную волну, а прямоугольный сигнал имеет только два состояния переключения (то есть высокий или низкий уровень наших цифровых контактов). Регулируя соотношение продолжительности включения и выключения, можно моделировать напряжение от 0 до 5 В. Затраченное время (академически называемое высоким уровнем) называется шириной импульса, поэтому ШИМ также называется широтно-импульсной модуляцией.

Давайте узнаем больше о ШИМ через следующие пять квадратных волн.



На приведенном выше рисунке зеленая линия представляет период, а значение `analogWrite()` соответствует проценту, который также называется рабочим циклом. Рабочий цикл подразумевает, что длительность высокого уровня делится на продолжительность низкого уровня в цикле. Сверху до



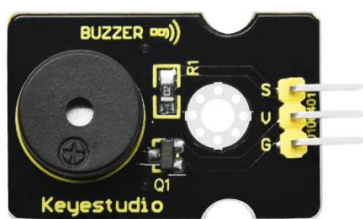
внизу, рабочий цикл первой прямоугольной волны равен 0%, а соответствующее ему значение - 0.

Яркость светодиода самая низкая, то есть выключен. Чем дольше длится высокий уровень, тем ярче светодиод. Следовательно, последний рабочий цикл равен 100%, что соответствует 255, светодиод самый яркий. 25% означает темнее.

ШИМ в основном используется для регулировки яркости светодиода или скорости вращения двигателя.

Проект 3: Пассивный зуммер

1. Описание



Arduino выполняет множество интерактивных работ.

Самый распространенный - это звуковая и световая индикация. Мы всегда используем светодиоды для экспериментов. Для этого урока мы разработаем схему, излучающую звук. Универсальные звуковые компоненты - зуммер и рожки. Зуммер проще в использовании. А зуммер включает в себя активный зуммер и пассивный зуммер. В этом эксперименте мы используем пассивный зуммер.

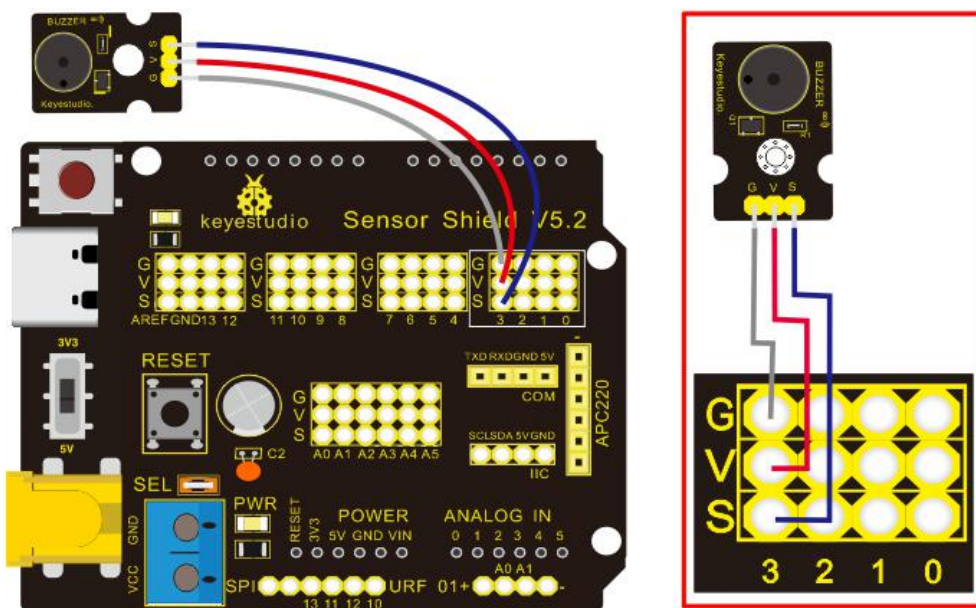
Используя пассивный зуммер, мы можем управлять разными звуками, вводя прямоугольные волны с разной частотой. Во время эксперимента мы контролируем код, который издает звуковой сигнал, начиная со звука «тик-тик», затем заставляем пассивный зуммер издавать «do re mi fa so la si do» и воспроизводить определенные песни.



2. Что вам нужно

PLUS Control	Датчик	Пассивный		3-контактный FF Dupont
Доска * 1	Щит * 1	Зуммер * 1	USB-кабель * 1	Кабель * 1
				

3. Схема подключения



Контакты G, V и S пассивного зуммер подключаются к G, V и 3.

4. Тестовый код



/ *

Комплект умного дома Keyestudio для Arduino

Project 3.1

Зуммер

<http://www.keyestudio.com>

* /

```
int tonepin = 3; // Установите контакт зуммера на цифровой D3
```

```
установка void ()
```

```
{
```

```
pinMode (tonepin, ВЫХОД); // Устанавливаем режим вывода цифрового ввода-вывода на вывод
```

```
}
```

```
пустой цикл ()
```

```
{
```

```
бесзнаковый символ i, j; в то
```

```
время как (1)
```

```
{
```

```
for (i = 0; i <80; i ++) // вывод частотного звука {
```

```
digitalWrite (tonepin, ВЫСОКИЙ); // Задержка
```

```
звука (1); // Задержка 1 мс
```

```
digitalWrite (tonepin, LOW); // Без задержки
```

```
звука (1); // Задержка 1 мс
```



```
}  
  
for (i = 0; i <100; i ++) // вывод звука другой частоты {  
  
    digitalWrite (tonepin, ВЫСОКИЙ); // Задержка  
  
    звука (2); // задержка 2 мс  
  
    digitalWrite (tonepin, LOW); // Без задержки  
  
    звука (2); // задержка 2 мс  
  
    }  
  
// *****  
  
*****
```

Из приведенного выше кода числа 80 и 100 определяют частоту в операторе «for». Время задержки контролирует продолжительность, как ритм в музыке.



Мы будем играть сказочную музыку, если будем хорошо контролировать частоту и ритм, поэтому давайте разберемся с частотой тонов. Как показано ниже:



Бас:

Тональная нота	1 #	2 #	3 #	4 #	5 #	6 #	7 #
A	221	248	278	294	330	371	416
B	248	278	294	330	371	416	467
C	131	147	165	175	196	221	248
D	147	165	175	196	221	248	278
E	165	175	196	221	248	278	312
F	175	196	221	234	262	294	330
грамм	196	221	234	262	294	330	371

Альта:



Тональная нота	1	2	3	4	5	6	7
A	441	495	556	589	661	742	833
B	495	556	624	661	742	833	935
C	262	294	330	350	393	441	495
D	294	330	350	393	441	495	556
E	330	350	393	441	495	556	624
F	350	393	441	495	556	624	661
грамм	393	441	495	556	624	661	742



Тональная нота	1 #	2 #	3 #	4 #	5 #	6 #	7 #
A	882	990	1112	1178	1322	1484	1665
B	990	1112	1178	1322	1484	1665	1869 г.
C	525	589	661	700	786	882	990
D	589	661	700	786	882	990	1112
E	661	700	786	882	990	1112	1248
F	700	786	882	935	1049	1178	1322
грамм	786	882	990	1049	1178	1322	1484

Далее нам нужно контролировать время воспроизведения ноты. Музыка будет воспроизводиться, когда каждая нота играет определенное количество времени. Ритм ноты делится на одну долю, половину доли, 1/4 доли, 1/8 доли ,.

Время для ноты определяется как половина доли (0,5), 1/4 доли (0,250, 1/8 долей).



beat (0,125) значит, играет музыка.

Возьмем пример «Оды радости».

Ode to joy

Beethoven

1=D $\frac{4}{4}$

3 3 4 5 | 5 4 3 2 | 1 1 2 3 | 3 . 2 2 - |
 Joy-ful, joy- ful, we a- dore thee, God of glo- ry, lord of love.

3 3 4 5 | 5 4 3 2 | 1 1 2 3 | 2 . 1 1 - |
 Heart un-flo-d like flowers be-fore thee, O-pen-ing to the sun a -bove.

||: 2 2 3 1 | 2 3 4 3 1 | 2 3 4 3 2 | 1 2 5 3 |
 Melt the clouds of sin and sad-ness. riue the dark of doubts a- way. Giv-

3 3 4 5 | 5 4 3 4 2 | 1 1 2 3 | 2 . 1 1 - :|
 -ver of im - mor-tal glad-ness, full us with the light of day.

Судя по нотной записи, музыка 4/4 удара. Есть особые

примечания, которые нам нужно объяснить:

1. Нормальная нота, как и первая нота 3, соответствует 350 (частота), занимает 1 долю
2. Нота с подчеркиванием означает 0,5 долей
3. Примечание с точкой (**3 .**) означает, что добавлено 0,5 удара, то есть 1 + 0,5 удара
4. Нота со знаком «-» означает, что добавляется 1 доля, то есть 1 + 1 доля.



5. Две последовательные ноты с дугой подразумевают легато, вы можете немного изменить частоту ноты после легато (необходимо отладить ее самостоятельно), например, уменьшив или увеличив некоторые значения, звук будет более плавным.

/ *

Комплект умного дома Keyestudio для Arduino

Project 3.2

Зуммер музыка

<http://www.keyestudio.com>

* /

определить NTD0 -1

определить NTD1 294

определить NTD2 330

определить NTD3 350

определить NTD4 393

определить NTD5 441

определить NTD6 495

определить NTD7 556

определить NTDL1 147

определить NTDL2 165

определить NTDL3 175

определить NTDL4 196



определить NTDL5 221

определить NTDL6 248

определить NTDL7 278

определить NTDH1 589

определить NTDH2 661

определить NTDH3 700

определить NTDH4 786

определить NTDH5 882

определить NTDH6 990

определить NTDH7 112

// Список всех D-настроенных частот

определить BCE 1

define HALF 0.5

define QUARTER 0.25

define EIGHTH 0.25

define SIXTEENTH 0.625 //

Список всех долей

int tune [] = // Список каждой частоты в соответствии с обозначением {

NTD3, NTD3, NTD4, NTD5,

NTD5, NTD4, NTD3, NTD2,



```
NTD1, NTD1, NTD2, NTD3,  
NTD3, NTD2, NTD2,  
NTD3, NTD3, NTD4, NTD5,  
NTD5, NTD4, NTD3, NTD2,  
NTD1, NTD1, NTD2, NTD3,  
NTD2, NTD1, NTD1,  
NTD2, NTD2, NTD3, NTD1, NTD2,  
NTD3, NTD4, NTD3, NTD1, NTD2,  
NTD3, NTD4, NTD3, NTD2, NTD1,  
NTD2, NTDL5, NTD0, NTD3, NTD3,  
NTD4, NTD5,  
NTD5, NTD4, NTD3, NTD4, NTD2,  
NTD1, NTD1, NTD2, NTD3,  
NTD2, NTD1, NTD1  
};  
  
float durt [] = // Список ударов в соответствии с обозначением {  
  
1,1,1,1,  
1,1,1,1,  
1,1,1,1,  
1 + 0.5,0.5,1 + 1,  
1,1,1,1,
```



```
1,1,1,1,  
1,1,1,1,  
1 + 0.5,0.5,1 + 1,  
1,1,1,1,  
1,0,5,0,5,1,1,  
1,0,5,0,5,1,1,  
1,1,1,1,  
1,1,1,1,  
1,1,1,0,5,0,5,  
1,1,1,1,  
1 + 0,5,0,5,1 + 1,};
```

```
int length;
```

```
int tonepin = 3; // Используем интерфейс
```

```
3 void setup ()
```

```
{
```

```
pinMode (tonepin, ВЫХОД);
```

```
длина = sizeof (мелодия) / sizeof (мелодия [0]); // Рассчитываем длину
```

```
}
```

```
пустой цикл ()
```

```
{
```

```
для (int x = 0; x <длина; x ++)
```



```
{
```

```
    тон (tonepin, мелодия [x]);
```

```
    задержка (350 * durt [x]); // Используется для настройки задержки в соответствии с ритмом, 350
```

можно настроить самостоятельно.

```
    noTone (тоновая шпилька);
```

```
}
```

```
    задержка (2000); // задержка 2S
```

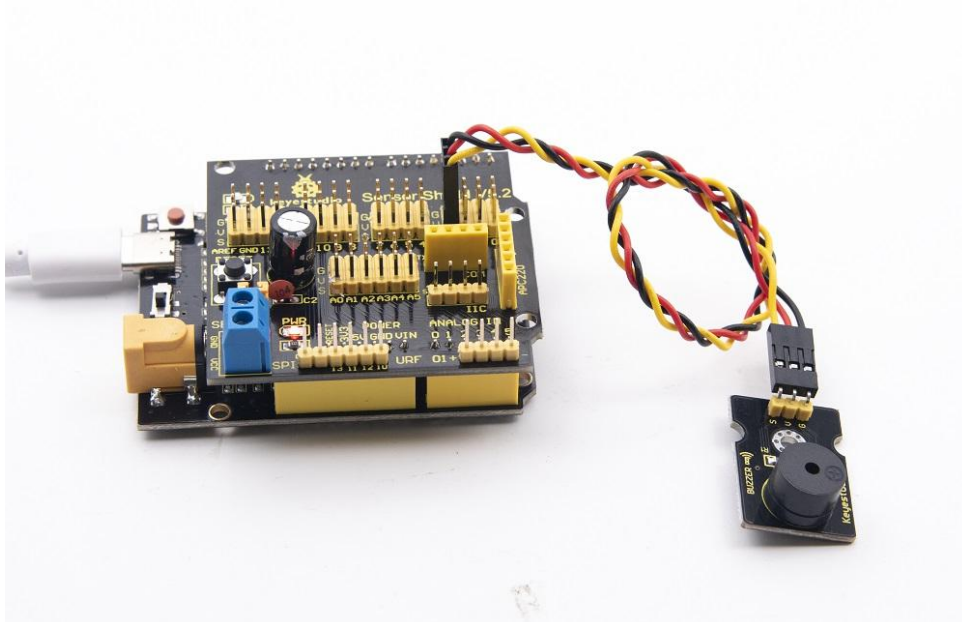
```
}
```

```
// *****
```

```
*****
```

Загрузите тестовый код на плату для разработки.

Вы слышите «Оду радости»?





Проект 4: Датчик кнопки

1. Описание

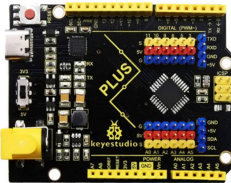
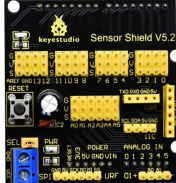
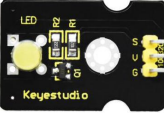



В этом уроке мы будем использовать функцию ввода порта ввода / вывода, то есть чтение выходного значения внешнего устройства. Кроме того, мы проведем эксперимент с кнопкой и светодиодом, чтобы узнать больше о вводе-выводе.

Кнопочный переключатель - это обычное дело в нашей жизни. Он относится к компонентам переключателя количества (цифрового количества). Состоящий из нормально разомкнутого контакта и нормально замкнутого контакта, он похож на обычный переключатель.

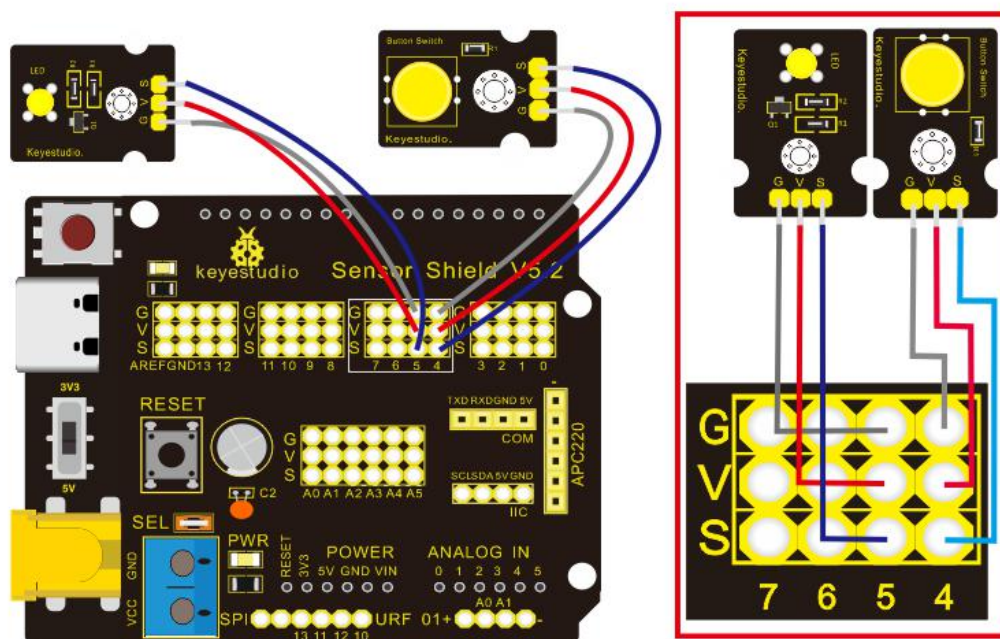
Когда нормально открытый контакт выдерживает давление, контур будет включен; однако, когда это давление исчезнет, нормально открытый контакт вернется в исходное состояние, то есть в выключенное состояние.

2. Что вам нужно



PLUS Control	Датчик	Желтый	Кнопка	USB	ЗpinF-F
Доска * 1	Щит * 1	ВЕЛ	Датчик * 1	Кабель * 1	Дюпон
		Модуль * 1			Кабель * 2
					

3. Схема подключения



Примечание. Контакты G, V и S сенсорного модуля кнопки отдельно подключаются к G, V и 4 на экране, а контакты G, V и S желтого светодиодного модуля связаны с G, V и 5 на экране.

5. Тестовый код

Затем мы разработаем программу, которая будет включать светодиод по кнопке. По сравнению с



В предыдущих экспериментах мы добавляли условное суждение --- оператор «если».

Письменные предложения Arduino основаны на языке C, поэтому формулировка условия суждения C подходит для Arduino, например while, switch и т. Д.

В этом уроке мы возьмем простую инструкцию «если» в качестве примера, чтобы продемонстрировать:

Если кнопка нажата, цифра 4 - низкий уровень, тогда мы делаем выход цифрового 5 высоким уровнем, тогда светодиод будет гореть; и наоборот, если кнопка отпущена, цифра 4 - высокий уровень, мы сделаем цифровой выход 5 низким уровнем, тогда светодиод погаснет.

Что касается вашей ссылки:

/ *

Комплект умного дома Keyestudio для Arduino

Project 4

Кнопка

<http://www.keyestudio.com>

* /

```
int ledpin = 5; // Определение светодиода в D5 int
```

```
inpin = 4; // Определяем кнопку в D4 int val; //
```

```
Определить переменную val
```

```
установка void ()
```

```
{
```




```
pinMode (ledpin, ВЫХОД); // Интерфейс светодиодной подсветки определяется как
output pinMode (inpin, INPUT); // Определяем интерфейс кнопки как ввод}

пустой цикл ()

{

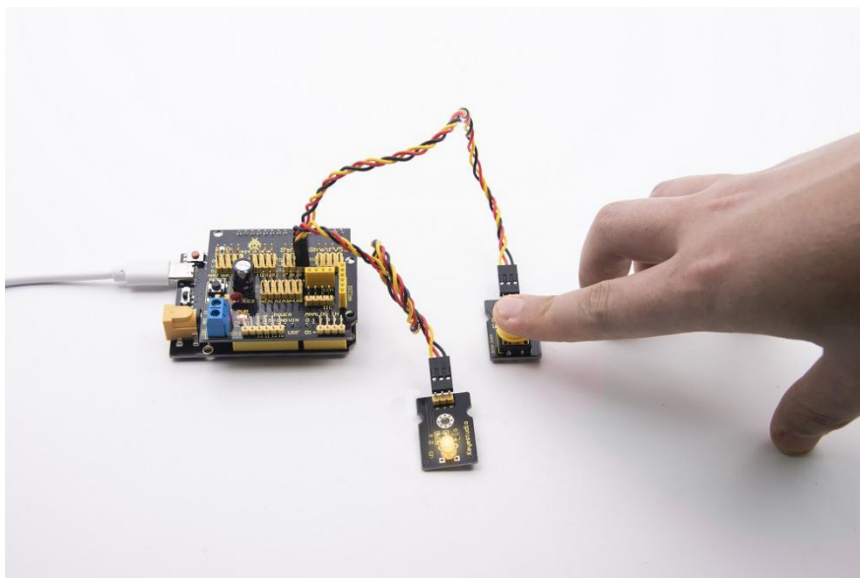
val = digitalRead (inpin); // Считываем цифровое значение 4 уровня и присваиваем его val
if (val == LOW) // Независимо от того, нажата ли клавиша, при нажатии будет гореть
свет {digitalWrite (ledpin, HIGH);}

еще
{digitalWrite (ledpin, LOW);}}

// *****
```

Этот эксперимент довольно прост и широко применяется к различным схемам и электрическим приборам.

Подсветка будет включаться при нажатии кнопки



Проект 5: 1-канальный релейный модуль



1. Описание:

Этот модуль представляет собой специальный модуль Arduino, совместимый с платой расширения датчика Arduino. Он имеет систему управления (также называемую входным контуром) и управляемую систему (также называемую выходным контуром).

Обычно используемый в схемах автоматического управления, релейный модуль представляет собой «автоматический переключатель», который управляет большим током и более низким напряжением с меньшим током и более низким напряжением.

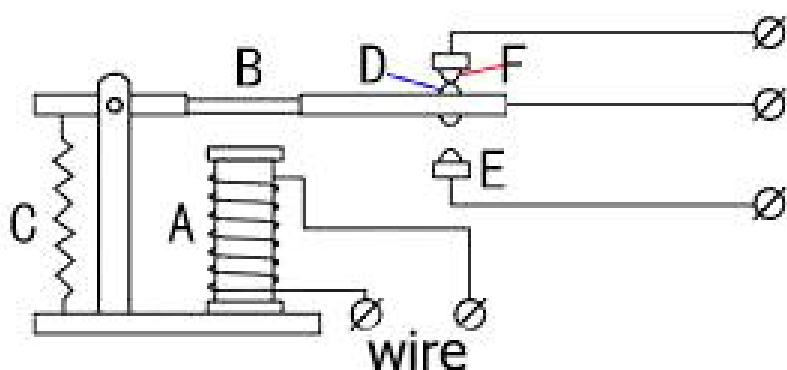
Таким образом, он играет роль автоматической регулировки, защиты и преобразования в цепи. Это позволяет Arduino управлять нагрузками ниже 3А, такими как светодиоды.



световые полосы, двигатели постоянного тока, миниатюрные водяные насосы, интерфейс электромагнитного клапана.

Основными внутренними компонентами релейного модуля являются электромагнит A, якорь.

B, пружина C, подвижный контакт D, статический контакт (нормально открытый контакт) E и статический контакт (нормально закрытый контакт) F (как показано на рисунке).



Пока к обоим концам катушки приложено определенное напряжение, через катушку будет течь определенный ток, вызывающий электромагнитные эффекты, и якорь будет притягивать железный сердечник против тянущей силы возвратной пружины под действием электромагнитной силы. притяжение, тем самым заставляя движущийся контакт и статический контакт (нормально открытый контакт) притягиваться. Когда катушка отключена, электромагнитное всасывание также исчезнет, и якорь вернется в исходное положение под действием силы реакции пружины, освобождая подвижный контакт и исходный статический контакт (нормально замкнутый контакт). Это втягивает и отпускает, таким образом достигая цели включения и выключения в цепи. «Нормально разомкнутые и замкнутые» контакты реле можно различить таким образом: статические контакты в отключенном состоянии, когда катушка реле отключена, называются «нормально разомкнутыми контактами»; статические контакты на подключенных



состояние называется «нормально замкнутый контакт». В модуле есть 2 установочных отверстия, чтобы вы могли закрепить модуль на другом оборудовании.

2. Характеристики:

Рабочее напряжение: 5 В (постоянный ток)

Входной сигнал: цифровой сигнал (высокий уровень 1, низкий уровень 0)

Контакты: статические контакты (нормально разомкнутые контакты, нормально замкнутые контакты)

и перемещение контактов

Номинальный ток: 10A (NO) 5A (NC)

Максимальное коммутируемое напряжение: 150 В (AC) 24 В (DC) Ток

поражения электрическим током: менее 3А

Время действия контакта: 10 мс

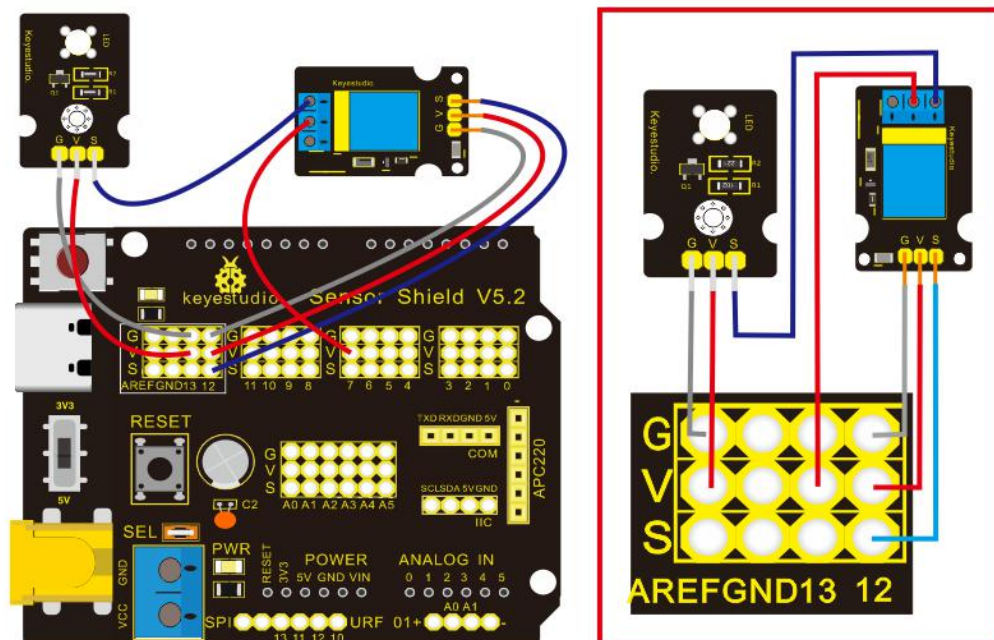
3. Что вам нужно

Плата управления PLUS * 1	Сенсорный экран * 1	USB-кабель * 1
		
Модуль реле * 1	Белый светодиодный модуль * 1	3-контактный кабель FF Dupont * 1



		
От женщины к женщине	От мужчины к женщине Dupont	
Кабели Dupont * 2	Кабели * 2	
		

4. Схема подключения:



Примечание. На экране контакты G, V и S 1-канального релейного модуля подключены к G, V и 12 соответственно. NO связан с V; контакты G, V и S белого светодиода соответственно подключены к G, V и статическому контакту NO на релейном модуле.



5. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 5

Реле

<http://www.keyestudio.com>

* /

int Relay = 12; // Определяем вывод реле на D12

void setup ()

{

pinMode (13, ВЫХОД); // Установить контакт 13 как выход

digitalWrite (13, HIGH); // Устанавливаем высокий уровень Pin13

pinMode (Relay, OUTPUT); // Установить Pin12 как выход}

пустой цикл ()

{

digitalWrite (реле, ВЫСОКИЙ); // Отключить

задержку реле (2000);

digitalWrite (реле, LOW); // Включаем реле

задержки (2000);

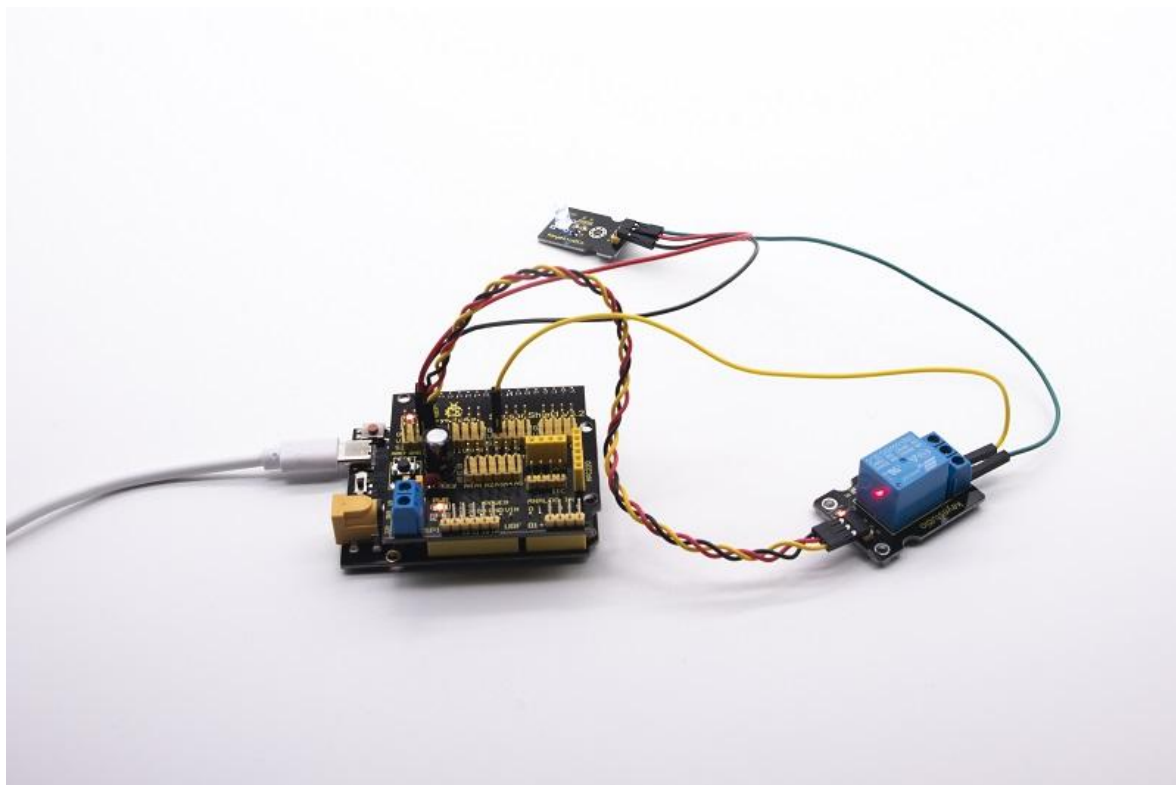
}



//*****

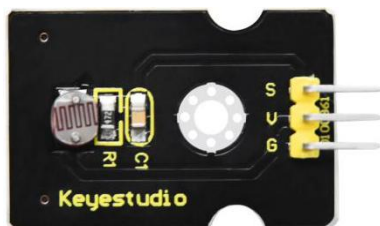
6. Результат теста:

Когда реле подключено («NO» включено, NC выключено) на 0,5 с, белый светодиод будет гореть; и наоборот, когда он отключен, белый светодиод погаснет.



Проект 6: Датчик фотоэлемента

1. Описание:



Датчик фотоэлемента (фоторезистор) представляет собой резистор, созданный за счет фотоэлектрического эффекта полупроводника. Поскольку он очень чувствителен к окружающему свету, его значение сопротивления меняется в зависимости от интенсивности света.

Его сигнальный конец подключен к аналоговому порту микроконтроллера. Когда интенсивность света увеличивается, сопротивление уменьшается, а аналоговое значение микроконтроллера - нет. Напротив, при уменьшении интенсивности света аналоговое значение микроконтроллера будет уменьшаться.

Таким образом, мы можем использовать модуль датчика фоторезистора для считывания соответствующего аналогового значения и определения интенсивности света в окружающей среде.

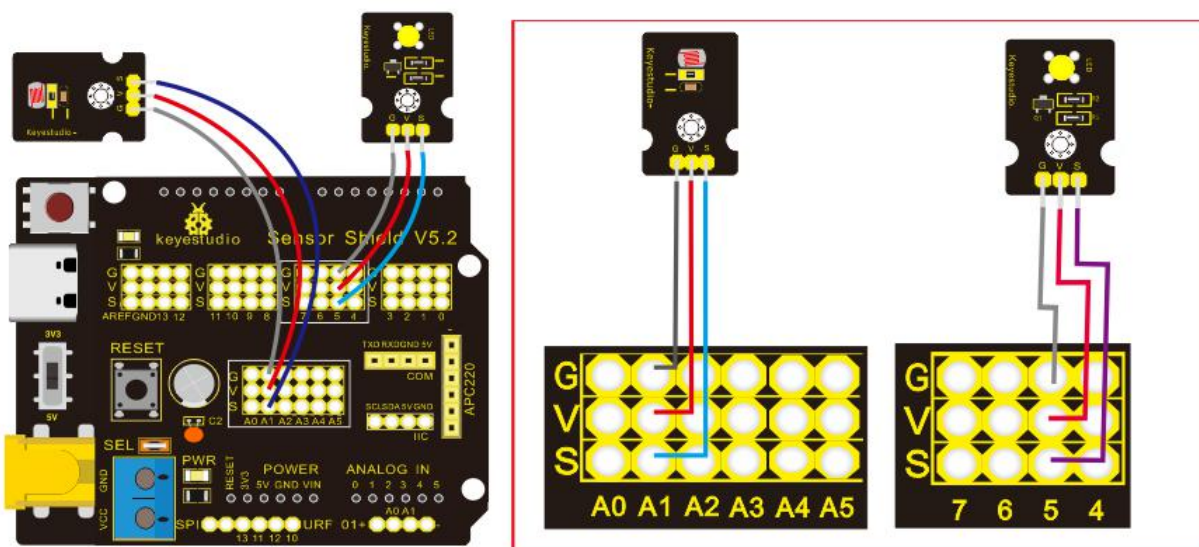
Это обычно измерение света, управление и преобразование, управление освещением применяется и к цепи.

2. Что вам нужно

Датчик управления PLUS	Фотоэлемент	Желтый светодиод	USB	3-контактный FF	
Доска * 1 Щит *	Датчик * 1	Модуль * 1	Кабель * 1	Дюпон	
					



3. Схема подключения:



Примечание. На плате расширения контакты G, V и S модуля датчика фотоэлемента подключены к G, V и A1; контакты G, V и S модуля желтого светодиода соединены с G, V и 5V отдельно.

4. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 6

фотоэлемент

<http://www.keyestudio.com>

* /

```
int LED = 5; // Устанавливаем вывод светодиода на D5
```



```
int val = 0; // Считываем значение напряжения фотодиода

void setup () {

    pinMode (светодиод, ВЫХОД); // светодиод выводится

    Serial.begin (9600); // Скорость передачи последовательного порта установлена на 9600

}

void loop () {

    val = analogRead (A1); // Считываем значение напряжения вывода A1 Serial.println (val); //

    Последовательный порт для просмотра изменения значения напряжения if (val <900)

    {// Меньше 900, светодиод не горит

    digitalWrite (LED, LOW);

    }

    еще

    {// В противном случае загорится

    светодиод digitalWrite (LED, HIGH);

    }

    задержка (10); // Задержка 10 мс

}

//*****
```

Светодиод будет гореть после загрузки тестового кода. Если вы используете фонарик, чтобы указать на фотоэлемент, светодиод автоматически выключится. Однако, если вы выключите фонарик, светодиод

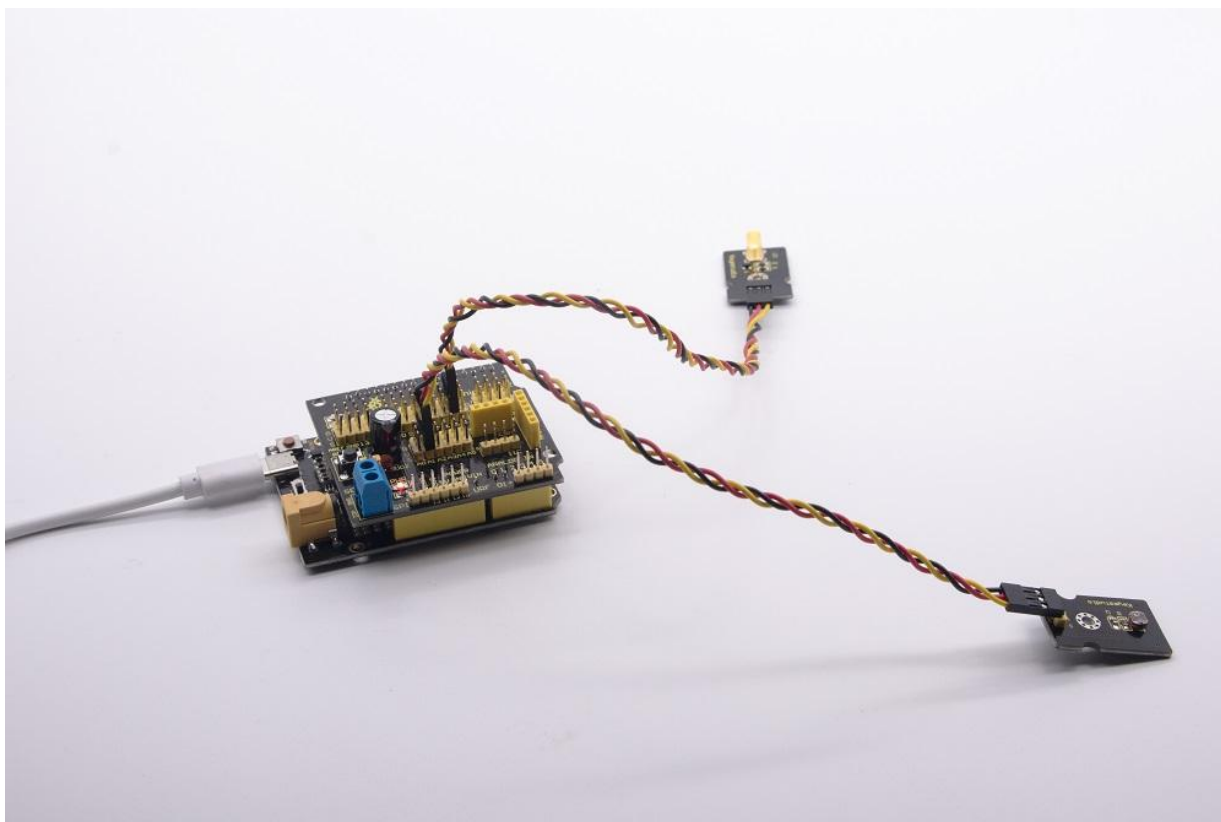


будет снова.

5. Обзор

Для этой строки кода это просто. Считываем значение через аналоговый порт и обращаем внимание на то, что аналоговая величина не требует режима ввода и вывода. Вы можете прочитать аналоговое значение датчика фотоэлемента через аналоговый порт.

Аналоговое значение будет постепенно уменьшаться, если есть свет. Когда значение до 900, это значение можно настроить в соответствии с выбранной вами яркостью.





Проект 7: Регулировка угла сервопривода двигателя



1. Описание:

Сервопривод может управлять дверьми и окнами. В этом курсе мы познакомимся с его принципом и продемонстрируем, как его использовать.

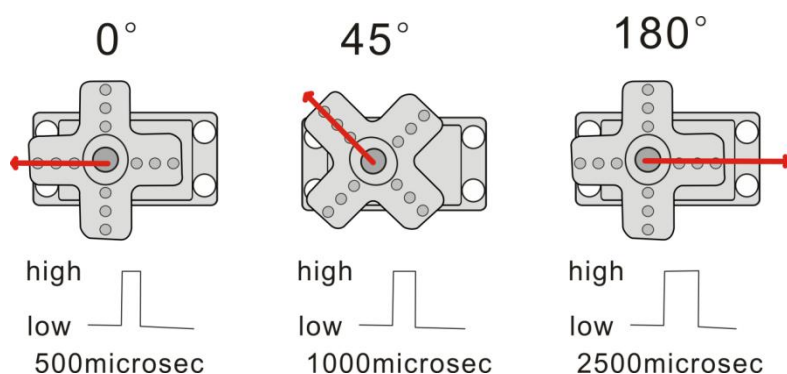
Серводвигатель - это поворотный привод с контролем положения. Он в основном состоит из корпуса, печатной платы, двигателя без сердечника, редуктора и датчика положения. Его принцип работы заключается в том, что сервопривод принимает сигнал, отправленный MCU или приемником, и производит опорный сигнал с периодом 20 мс и шириной 1,5 мс, затем сравнивает полученное напряжение смещения постоянного тока с напряжением потенциометра и получает выходной сигнал разности напряжений.

Когда скорость двигателя постоянна, потенциометр приводится во вращение через каскадный редуктор, что приводит к нулевой разнице напряжений, и двигатель перестает вращаться. Как правило, диапазон углов поворота сервопривода составляет 0° - 180°

Угол поворота серводвигателя регулируется путем регулирования рабочего цикла сигнала ШИМ (широтно-импульсной модуляции). Стандартный цикл сигнала ШИМ составляет 20 мс (50 Гц). Теоретически ширина распределяется от 1 до 2 мсек, но на самом деле это от 0,5 до 2,5 мс. Ширина соответствует углу поворота от 0° до 180° . Но обратите внимание, что для двигателей разных марок один и тот же сигнал может иметь разные



угол поворота.



Один из способов - использовать общий порт цифрового датчика Arduino для генерации прямоугольной волны с разным рабочим циклом, а также для имитации сигнала ШИМ и использования этого сигнала для управления позиционированием двигателя.

Другой использует функцию сервопривода Arduino для управления двигателем. Таким образом, программу будет проще разработать, но она сможет управлять только двухканальным двигателем, потому что функция сервопривода использует только цифровые выводы 9 и 10.

Емкость диска Arduino ограничена. Поэтому, если вам нужно управлять более чем одним двигателем, вам понадобится внешний источник питания.

Обратите внимание, что не подавайте питание через USB-кабель, есть возможность повредить USB-кабель, если текущая потребность превышает 500 мА. Мы рекомендуем внешний источник питания.

2. Технические характеристики:



Рабочее напряжение: 4,8 В ~ 6 В постоянного тока

Диапазон рабочих углов: около 180 ° (при 500 → 2500 мкс)

Диапазон ширины импульса: 500 → 2500 мкс

Скорость холостого хода: 0,12 ± 0,01 с / 60 (4,8 В постоянного тока) 0,1 ± 0,01 с / 60 (6 В постоянного тока)

Ток холостого хода: 200 ± 20 мА (4,8 В постоянного тока) 220 ± 20 мА (6 В постоянного тока)

Момент остановки: 1,3 ± 0,01 кг · см (4,8 В постоянного тока) 1,5 ± 0,1 кг · см (6 В постоянного тока)

Остановить ток: ≤ 850 мА (4,8 В постоянного тока) ≤ 1000 мА (6 В постоянного тока)


Ток в режиме ожидания: 3 ± 1 мА (4,8 В постоянного тока) 4 ± 1 мА (6 В постоянного тока)

Длина провода: 250 ± 5 мм

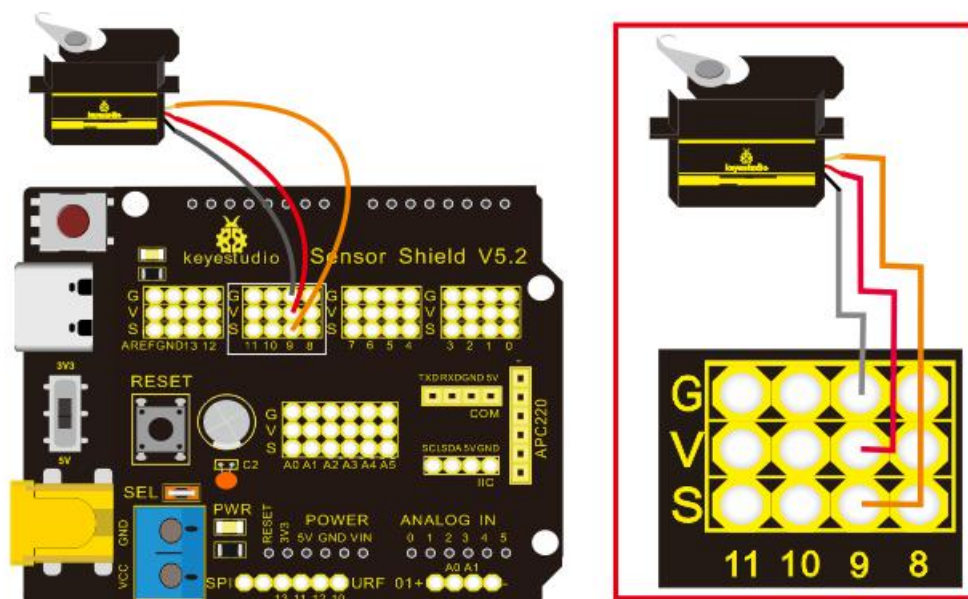
Размер внешнего вида: 22,9 * 12,2 * 30 мм

Вес: 9 ± 1 г (без сервопривода)

3. Что вам нужно

PLUS Control Доска * 1	Сенсорный экран * 1	Сервопривод * 1	USB-кабель * 1
			

4. Схема подключения:



Примечание: сервопривод подключен к G (GND), V (VCC), 9. Коричневый провод сервопривода подключен к Gnd (G), красный провод подключен к 5 В (V), а оранжевый провод подключен к цифровому выводу 9.

5. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 7

Серво

<http://www.keyestudio.com>

* /



```
# include <Servo.h> // Библиотека серво-функций

Servo myservo;

int pos = 0; // Начальный угол servo

void setup ()

{

myservo.attach (9); // Определяем положение сервопривода на D9}


пустой цикл ()

{

for (pos = 0; pos <180; pos + = 1) // угол от 0 до 180 градусов {

myservo.write (pos); // Угол сервопривода - pos

delay (15); // Задержка 15 мс

}

for (pos = 180; pos> = 1; pos- = 1) // Угол от 180 до 0 градусов {

myservo.write (pos); // Угол сервопривода - pos delay

(15); // Задержка 15 мс

}

}

//*****

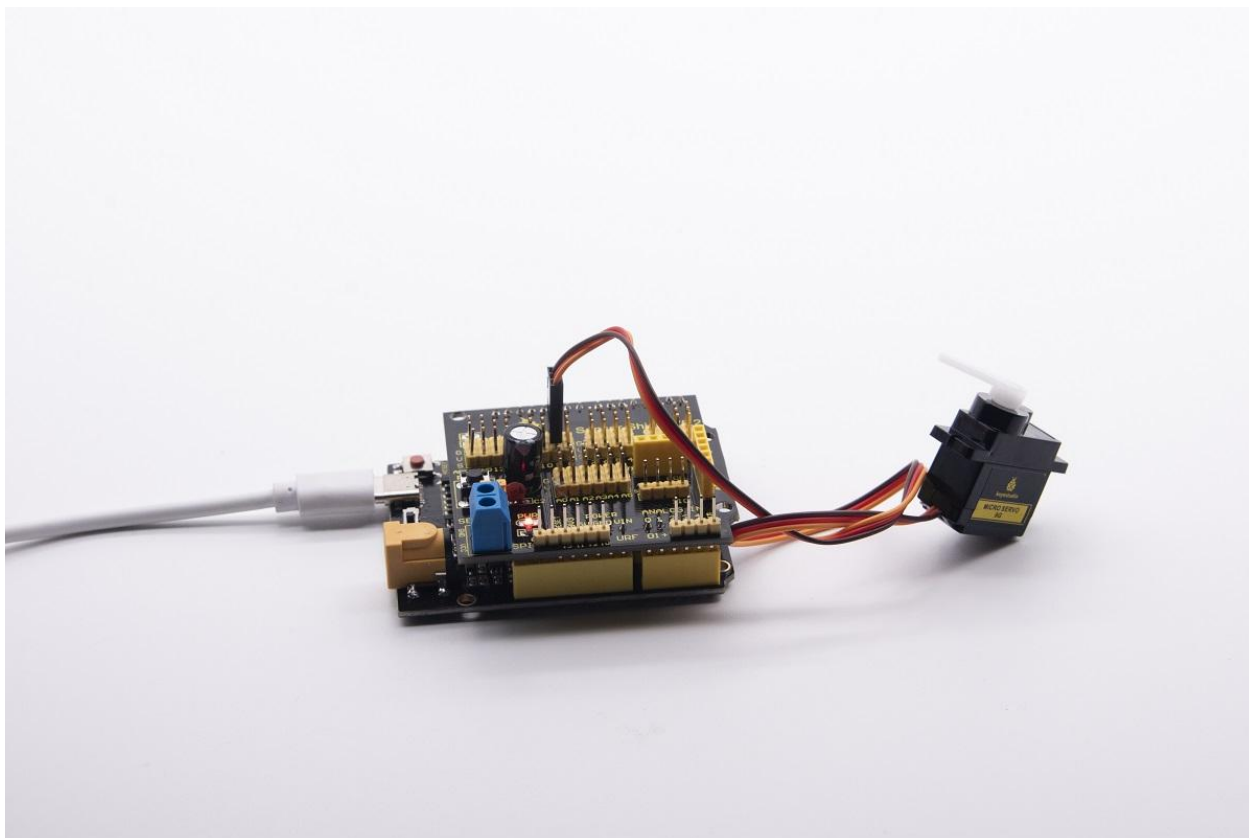
*****

*****
```




6. Результат теста:

Загрузите код, подключите компоненты в соответствии со схемой подключения и включите питание. Сервопривод вращается от 0 ° до 180 °, затем от 180 ° до 0 °.



Проект 8: Модуль вентилятора



1. Описание



Модуль вентилятора L9110 использует микросхему управления двигателем L9110 и контролирует направление вращения и скорость двигателя. Кроме того, этот модуль эффективен, с высококачественным вентилятором, который может погасить пламя на расстоянии 20 см. Точно так же это также важная часть пожарного робота.

2. Технические характеристики:

Рабочее напряжение: 5 В

Рабочий ток: 0,8 А

Совместимость выходного уровня TTL / CMOS,

Управление и привод интегрированы в ИС

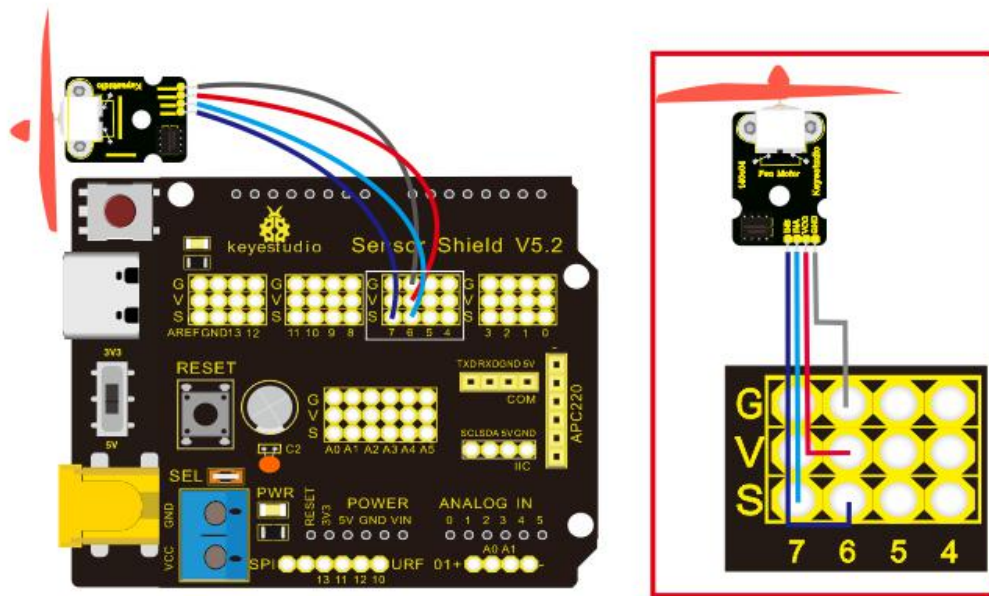
Имеют функцию защиты штифта от высокого давления

Рабочая температура: 0-80 °

3. Что вам нужно

PLUS Control	Датчик	Поклонник	USB	От женщины к женщине
Доска * 1	Щит * 1	Модуль * 1	Кабель * 1	Кабели Dupont * 4
				

4. Схема подключения:



Примечание. На экране контакты GND, VCC, INA и INB модуля вентилятора соответственно подключены к G, V, 7, 6.

5. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 8

Поклонник

<http://www.keyestudio.com>

* /

```
void setup () {
```

```
    pinMode (7, ВЫХОД); // определяем вывод D7 как выход
```



```
pinMode (6, ВЫХОД); //определять      Вывод D6 как выход

}

void loop () {

    digitalWrite (7, LOW);

    digitalWrite (6, ВЫСОКИЙ); // Обратное вращение мотора

    delay (3000); // задержка 3S

    digitalWrite (7, LOW);

    digitalWrite (6, LOW); // Двигатель перестает вращаться

    delay (1000); // задержка 1S

    digitalWrite (7, ВЫСОКИЙ);

    digitalWrite (6, LOW); // Двигатель вращается в прямом направлении

    delay (3000); // задержка 3S

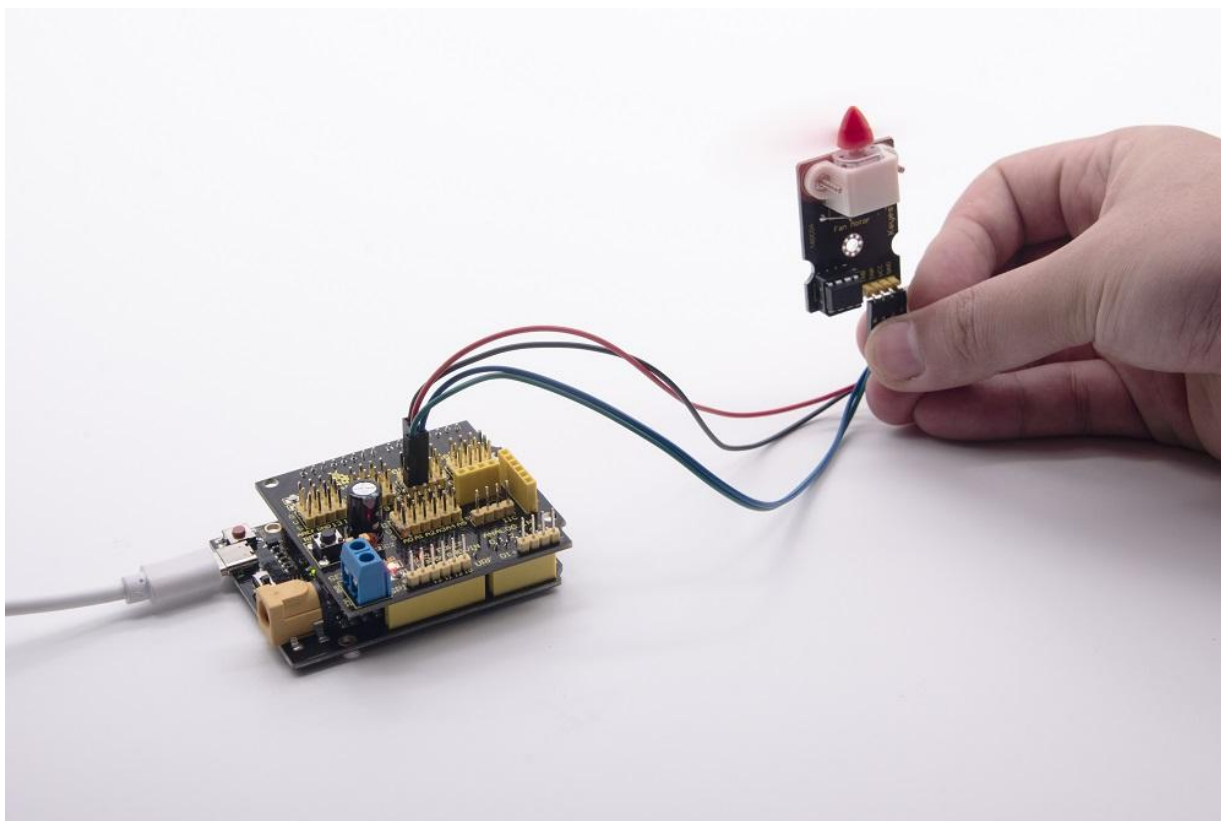
}

//*****

*****
```

6. Результат теста:

Загрузите тестовый код, подключите компоненты в соответствии со схемой подключения, поверните DIP-переключатель вправо и включите питание. Вентилятор вращается против часовой стрелки в течение 3000 мс, останавливается на 1000 мс, затем вращается по часовой стрелке в течение 3000 мс.



Проект 9: Датчик пара



1. Описание:

Это обычно используемый датчик пара. Его принцип состоит в том, чтобы определять количество воды по неизолированным параллельным линиям на печатной плате. Чем больше воды, тем больше будет подключено проводов. Как проводящее контактное покрытие



увеличивается, выходное напряжение будет постепенно расти. Он также может обнаруживать водяной пар в воздухе. Датчик пара можно использовать как датчик дождевой воды и реле уровня. Когда влажность на поверхности датчика резко возрастает, выходное напряжение увеличивается.

Датчик совместим с различными платами управления микроконтроллеров, такими как микроконтроллеры серии Arduino. При его использовании мы предоставляем руководство по работе с датчиком пара и платой управления Arduino.

Сначала подключите датчик к аналоговому порту микроконтроллера и отобразите соответствующее аналоговое значение на последовательном мониторе.

Примечание: соединительная часть не является водонепроницаемой, поэтому, пожалуйста, не погружайте ее в воду.

2. Технические характеристики:

Рабочее напряжение: 3,3-5 В постоянного тока

Рабочий ток: <20 мА

Диапазон рабочих температур: -10 °C ~ + 70 °C;

Управляющий сигнал: аналоговый выходной сигнал

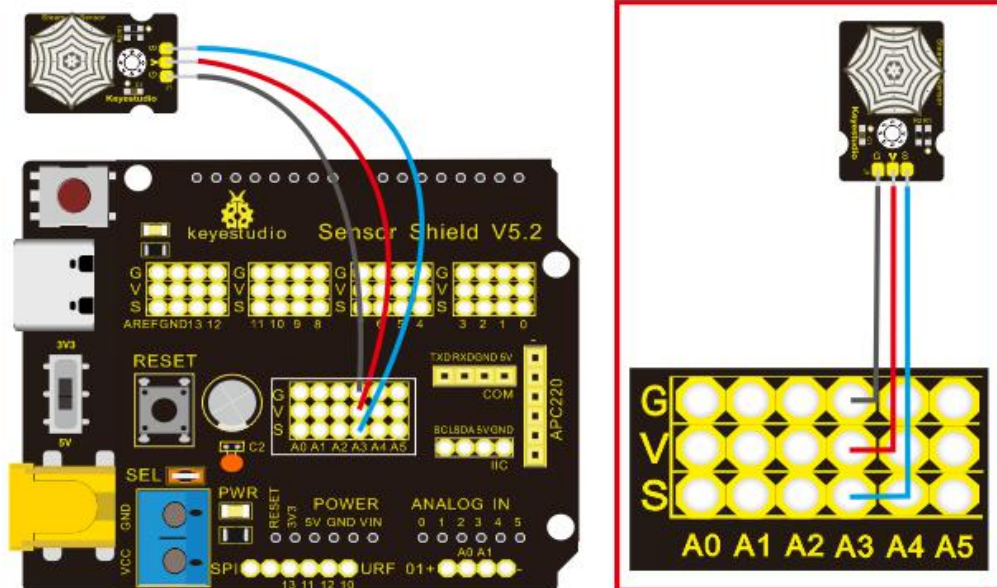
Интерфейс: 3-контактный интерфейс с шагом 2,54 мм



4. Что вам нужно

PLUS Control	Датчик	Готовить на пару	USB-кабель * 1	3-контактный FF Dupont
Доска * 1	Щит * 1	Датчик * 1		Кабель * 1
				

5. Схема подключения:



Примечание. На экране датчика контакты G, V и S датчика пара подключаются к G, V и A3



6. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 9

Готовить на пару

<http://www.keyestudio.com>

* /

установка void ()

{

Serial.begin (9600); // открываем последовательный порт и устанавливаем скорость передачи 9600 бит / с}

пустой цикл ()

{

int val;

val = analogRead (3); // подключаем датчик паров к аналоговому

порту 3 Serial.print ("Moisture is");

Serial.println (val, DEC); // считываем аналоговое значение через последовательный порт print

delay (100); // задержка 100 мс

}

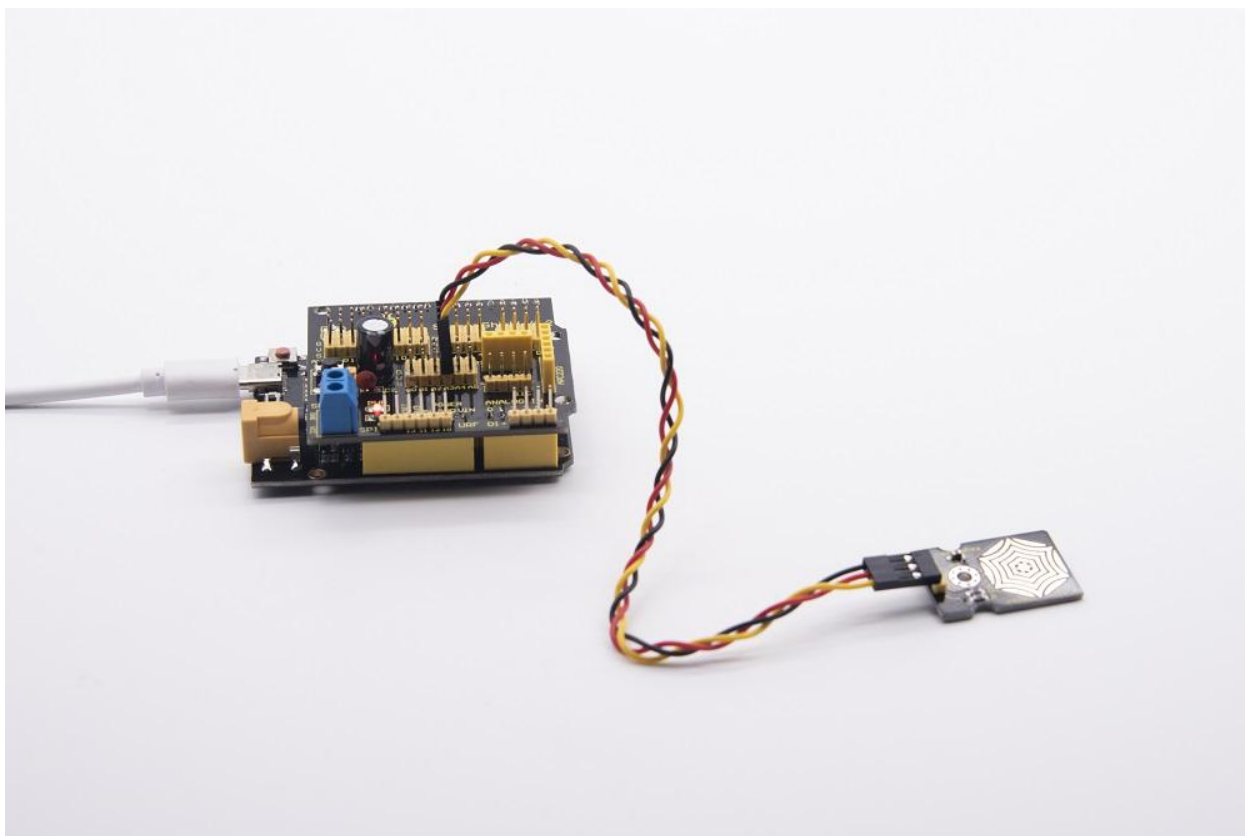
// *****

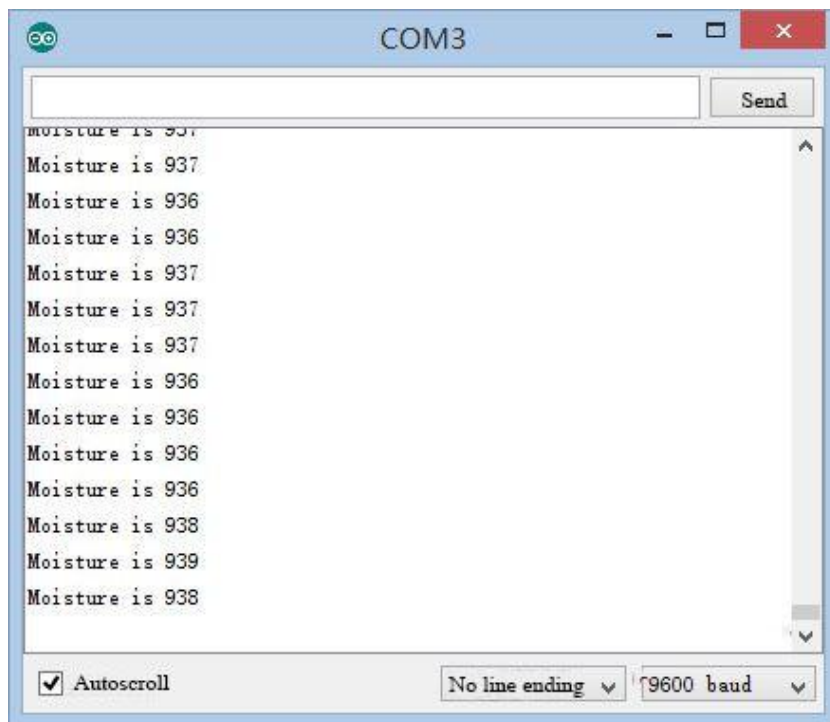


7. Результат теста:

При обнаружении различной влажности датчик будет получать обратную связь с другим значением тока. Как показано ниже;

Когда датчик обнаруживает пар кипяченой воды, значение влажности отображается на последовательном мониторе программного обеспечения ARDUINO.





Проект 10: Датчик движения PIR



1. Описание:

Пирозлектрический инфракрасный датчик движения может обнаруживать инфракрасные сигналы от движущихся объектов и выводить сигналы переключения. Применяемый в различных случаях, он может обнаруживать движение человеческого тела.

Обычные пирозлектрические инфракрасные датчики намного больше, со сложной схемой и меньшей надежностью. Тем не менее, этот новый пирозлектрический инфракрасный датчик движения более практичен.

Он объединяет цифровой пирозлектрический инфракрасный датчик и соединительные штыри. Отличается повышенной чувствительностью и надежностью, меньшим энергопотреблением, легкостью



вес, небольшие размеры, режим работы с низким напряжением и более простая периферийная схема.

2. Технические характеристики:

Входное напряжение: 3,3 В ~ 18 В

постоянного тока Рабочий ток: 15 мкА

Рабочая температура: -20 ~ 85 градусов Цельсия Выходное

напряжение: высокое 3 В, низкое 0 В

Время задержки вывода (высокий уровень): от 2,3 до 3 секунд

Угол обнаружения: около 100 °

Расстояние обнаружения: 3-4 метра

Светодиодный индикатор выхода (высокий

уровень) Предельный ток вывода: 100 мА

Примечание:

1. Максимальное расстояние во время тестирования составляет 3-4 метра.

2. Во время теста откройте белую линзу, чтобы проверить прямоугольную чувствительную часть.

Когда длинная линия чувствительной части параллельна земле, расстояние является наилучшим.

3. Во время теста, накрыв сенсор белой линзой, можно точно определить расстояние.

4. Расстояние лучше всего при 25 °C, и значение расстояния обнаружения уменьшится.



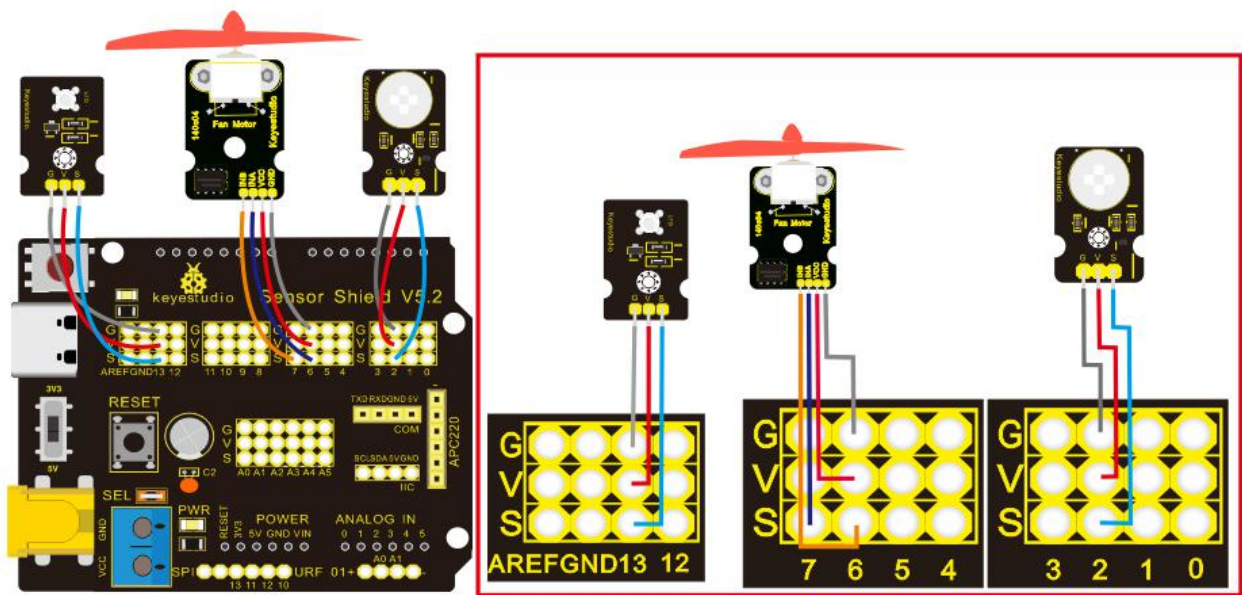
когда температура превышает 30 °C.

5. После включения и загрузки кода вы можете начать тестирование через 5-10 секунд, иначе датчик не чувствителен.

3. Что вам нужно

PLUS Control Доска * 1	Датчик Щит * 1	PIR движение Датчик * 1	От женщины к женщине Кабели Dupont * 4
			
Поклонник Модуль * 1	Белый светодиод Модуль * 1	USB-кабель * 1	3pinF-F Dupont Строка * 2
			

4. Схема подключения:



Примечание. На экране G, V и S датчика движения PIR подключены к G, V и 2; GND, VCC, INA и INB модуля вентилятора отдельно связаны с Г, В, 7,6. Контакты G, V и S светодиодного модуля связаны с G, V и 13.

5. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 10

PIR

<http://www.keyestudio.com>

* /

```
void setup () {
```



```
Serial.begin (9600); // открываем последовательный порт и устанавливаем скорость передачи
```

```
9600 бит / с pinMode (2, INPUT); // Определить PIR как вход в D2
```

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

```
pinMode (13, ВЫХОД); // Определить светодиод как выход в D13
```

```
pinMode (7, OUTPUT); // Определить D7 как вывод pinMode (6,
```

```
OUTPUT); // Определить D6 как вывод
```

```
}
```

```
void loop () {
```

```
Serial.println (digitalRead (2));
```

```
задержка (500); // Задержка 500 мс
```

```
if (digitalRead (2) == 1) // Если кто-то обнаружен идущим
```

```
{
```

```
digitalWrite (13, ВЫСОКИЙ); // Светодиод горит
```

```
digitalWrite (7, HIGH);
```

```
analogWrite (6 150); // Вентилятор вращается
```

```
} else // Если идущего человека не обнаружено
```

```
{
```

```
digitalWrite (13, LOW); // Светодиод не горит
```

```
digitalWrite (7, LOW);
```

```
analogWrite (6,0); // Вентилятор не вращается
```



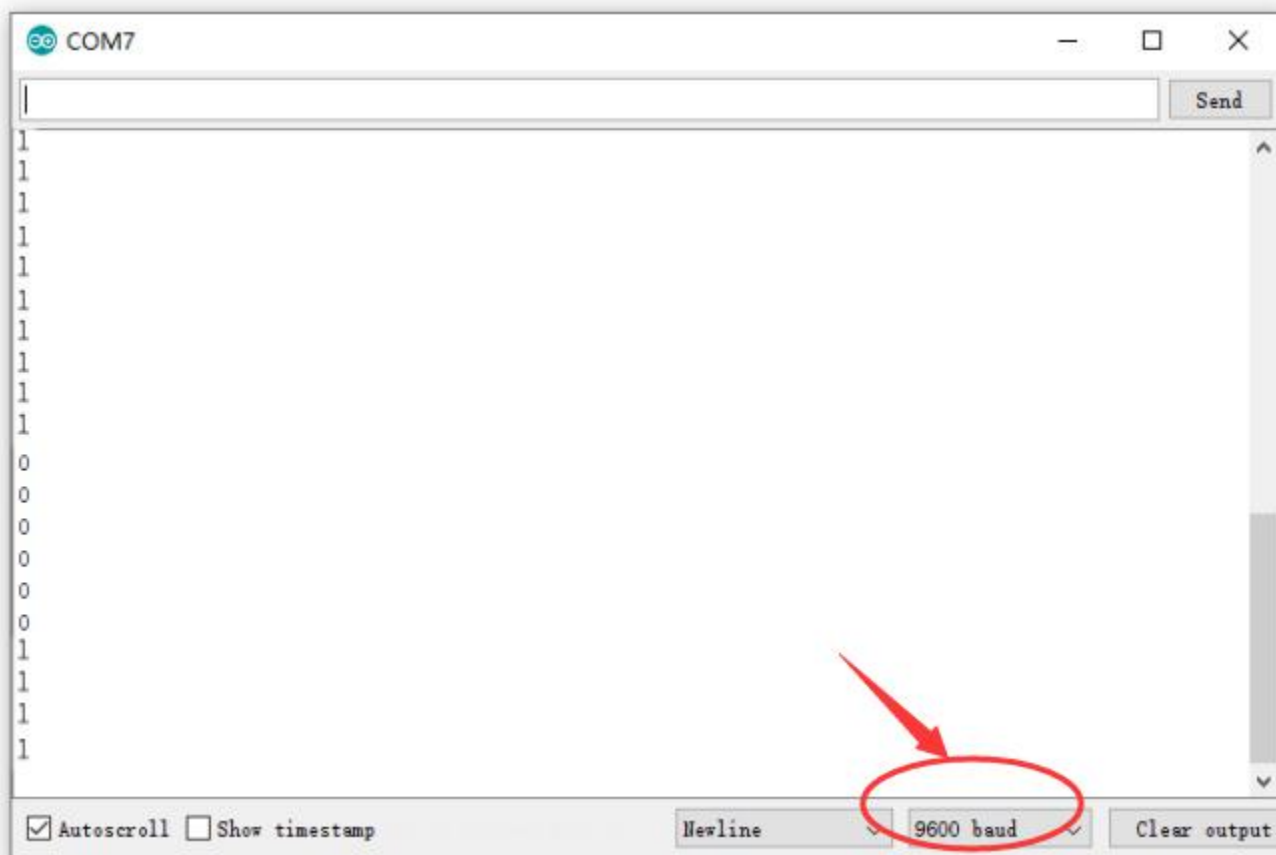
}}

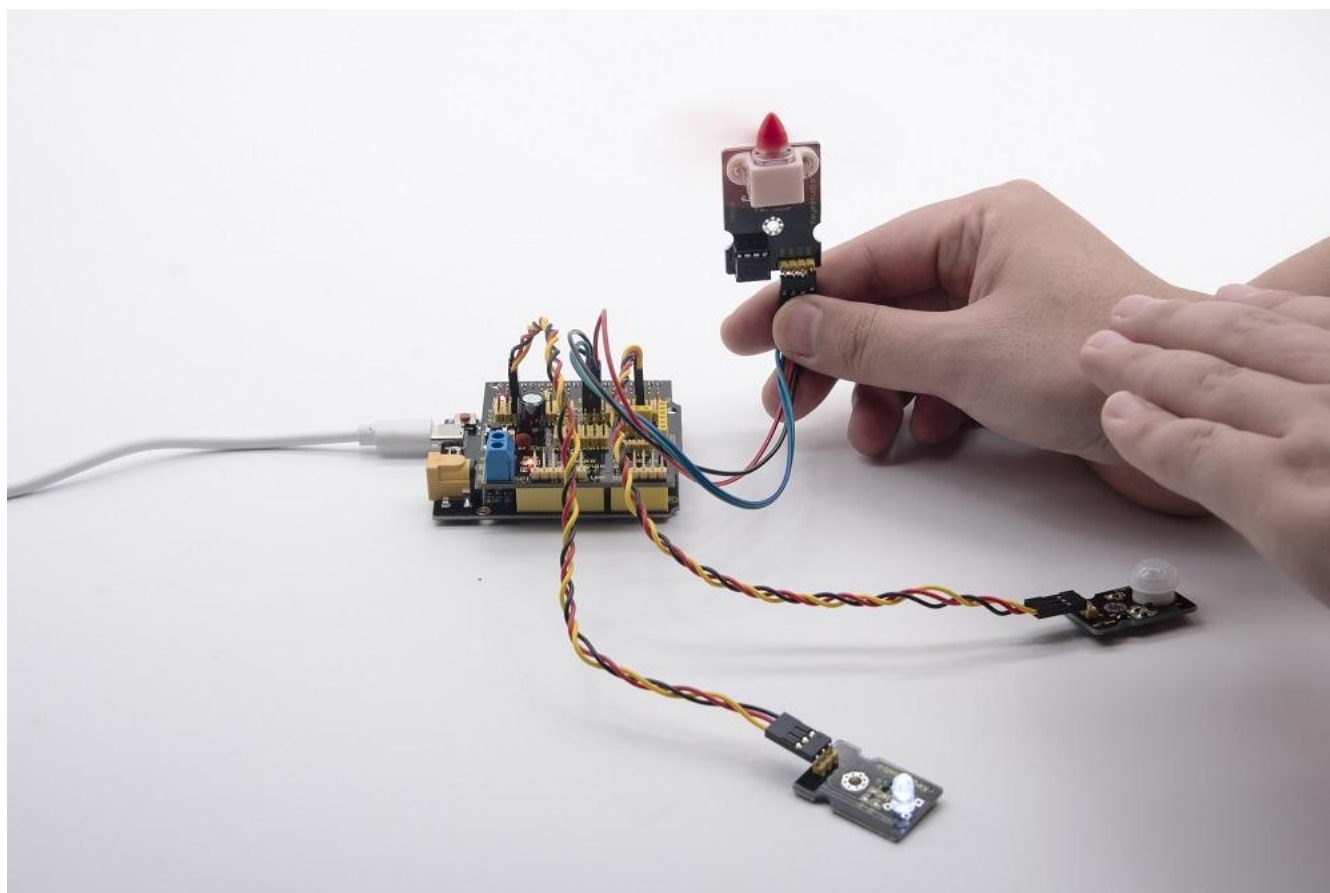
```
//*****
```

```
*****
```

6. Результат теста:

Загрузите приведенный выше тестовый код, откройте монитор последовательного порта и установите скорость передачи на 9600. Если датчик движения PIR обнаружит кого-то поблизости, монитор последовательного порта отобразит «1», а также загорятся светодиод и D13, и вентилятор будет вращаться. Если рядом никого нет, серийный монитор покажет «0», индикаторы погаснут и вентилятор перестанет вращаться.





Проект 11: Аналоговый газ (MQ-2) Датчик



1. Описание:

Этот датчик газа используется для сигнализации утечки газа в быту, сигнализации для промышленных горючих газов и портативных приборов обнаружения газа. Также он подходит для обнаружения сжиженного газа, бензола, алкана, спирта, водорода и т. Д.



Датчик дыма MQ-2 может быть детектором нескольких газов с такими преимуществами, как высокая чувствительность, быстрый отклик, хорошая стабильность, долгий срок службы и простая схема привода.

Он может обнаруживать концентрацию горючего газа и дыма в диапазоне 300 ~ 10000 ppm. Между тем, он имеет высокую чувствительность к природному газу, сжиженному углеводородному газу и другому дыму, особенно к дыму алканов.

Перед использованием датчика дыма его необходимо нагреть в течение некоторого времени, в противном случае выходное сопротивление и напряжение будут неточными. Однако напряжение нагрева не должно быть слишком высоким, в противном случае это приведет к перегоранию внутренней сигнальной линии.

Он относится к полупроводниковому газочувствительному материалу из диоксида олова. При определенной температуре диоксид олова адсорбирует кислород в воздухе и формирует адсорбцию отрицательных ионов кислорода, уменьшая электронную плотность в полупроводнике, тем самым увеличивая значение его сопротивления.

При контакте с горючим газом в воздухе и смогом, а потенциальный барьер на границе зерен регулируется смогом, это вызовет изменение поверхностной проводимости. Таким образом можно получить информацию о наличии дыма или горючего газа. Чем выше концентрация дыма или горючего газа в воздухе, тем выше проводимость и чем ниже выходное сопротивление, тем больше аналоговый выходной сигнал. Кроме того, чувствительность можно регулировать вращением потенциометра.



2. Технические характеристики:

Рабочее напряжение: 3,3-5 В (постоянный ток)

Интерфейс: 4 контакта (VCC, GND, D0, A0)

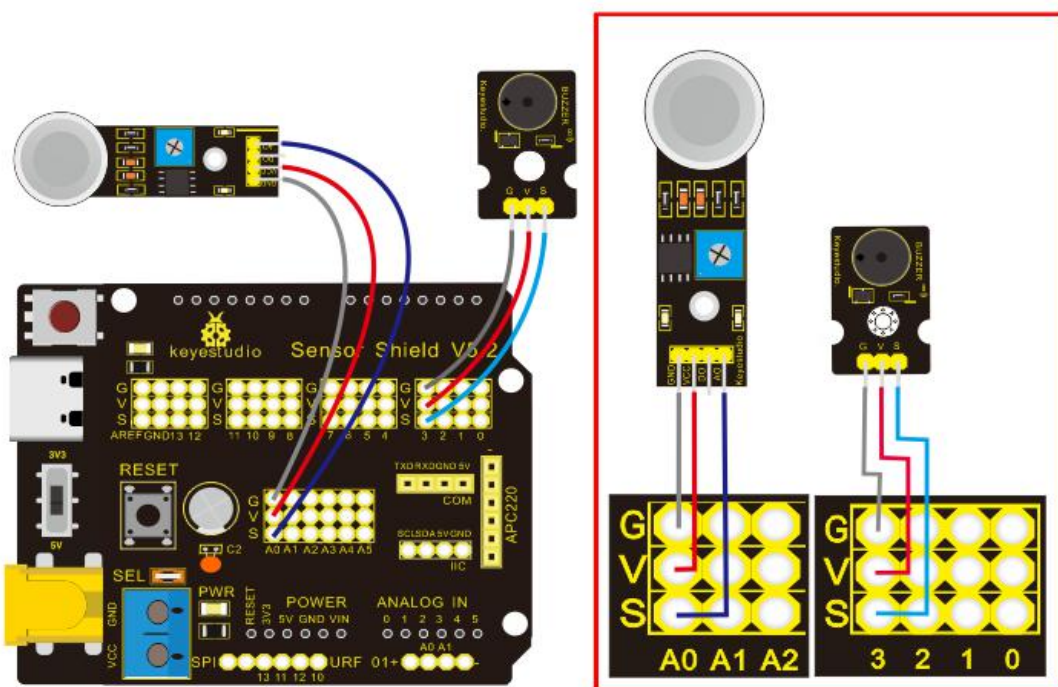
Выходной сигнал: цифровой сигнал и аналоговый сигнал

Вес: 7,5 г

3. Что вам нужно

ПЛЮС контроль Доска * 1	Датчик Щит * 1	MQ-2 Газ Датчик * 1	3pinF-F Dupont Кабель * 1
			
Пассивный зуммер * 1	USB-кабель * 1	Ф. Ф. Дюпон Кабель * 3	
			

4. Схема подключения:



Примечание. На экране контакты GND, VCC, D0 и A0 газового датчика связаны с контактами G, V и A0. Контакты G, V и S пассивного зуммера подключены к G, V и 3.

5. Тестовый код:

/ *

Комплект умного дома Keyestudio для Arduino

Project 11

Газ

<http://www.keyestudio.com>

* /

```
int MQ2 = A0; // Определить вывод датчика газа MQ2 на A0 int
```

```
val = 0; // объявляем переменную
```



```
int buzzer = 3; // Определяем вывод зуммера в D3

void setup ()

{

  pinMode (MQ2, ВХОД); // Датчик газа MQ2 в качестве входных данных Serial.begin (9600); //

  Устанавливаем скорость передачи последовательного порта на 9600 pinMode (buzzer, OUTPUT); //

  Устанавливаем режим вывода цифрового ввода-вывода для вывода}

пустой цикл ()

{

  val = analogRead (MQ2); // Считываем значение напряжения порта A0 и присваиваем его значению val

  Serial.println (val); // Последовательный порт отправляет значение val

  если (val> 450)

  {

    тон (зуммер, 589);

    задержка (300);

  }

  еще

  {

    noTone (зуммер);

  }

}

//***** *
```