

Супер обучающий комплект для Arduino

Содержание

1. Введение.....	13
2. Список компонентов	14
3.Установите Arduino IDE и драйвер	21 год
(1) Установка Arduino IDE	21 год
(2) плата разработки keyestudio V4.0	23
(3) Плата Keyestudio MEGA 2560	26
(4) Установка драйвера платы V4.0	29
(5) Настройка Arduino IDE	35 год
(6) Запустить первую программу	39
4.Как добавить библиотеку?	43
5. Детали проекта	47

Проект 1: Hello World	47
1. Введение.....	47
2. Требуемое оборудование	47
3. Образец кода	48
4. Результат теста	49
Проект 2: мигание светодиода	50
1. Введение.....	50
2. Требуемое оборудование	50
3. Незнание	51
4. Схема подключения	54
5. Образец кода	57
6. Результат теста	58
Проект 3: ШИМ	59
1. Введение.....	59
2. Принцип работы	59
3. Необходимое оборудование	61
4. Схема подключения	61
5. Образец кода	63
6. Результат теста	64
Проект 4: Светофор	65
1. Введение.....	65
2. Требуемое оборудование	66

3. Схема подключения	66
4..Пример кода	68
5. Результат теста	70
Проект 5: Эффект преследования светодиодов	71
1. Введение.....	71
2. Требуемое оборудование	72
3. Схема подключения	72
4. Образец кода	75
Светодиодный эффект погони	75
5. Результат теста	77
Проект 6: Светодиод с кнопочным управлением	78
1. Введение.....	78
2. Требуемое оборудование	78
3. Незнание	79
4. Схема подключения	81 год
5. Образец кода	83
6. Результат теста	84
Проект 7: Активный зуммер	86
1. Введение.....	86
2. Требуемое оборудование	86
3. Незнание	87
4. Схема подключения	87

5. Образец кода	89
6. Результат теста	90
Проект 8: Пассивный зуммер	91
1. Введение.....	91
2. Требуемое оборудование	92
3. Незнание	92
4. Схема подключения	93
5. Образец кода	94
6. Результат теста	96
Проект 9: светодиод RGB	97
1. Введение.....	97
2. Требуемое оборудование	97
3. Незнание	97
4. Схема подключения	98
5. Образец кода	100
6. Результат теста	102
Проект 10: Фоторезистор	103
1. Введение.....	103
2. Требуемое оборудование	103
3. Незнание	104
4. Схема подключения	105
5. Образец кода	107

6. Результат теста	108
 Проект 11: Датчик пламени	
1. Введение.....	109
2. Требуемое оборудование	110
3. Незнание	110
4.Принцип эксперимента	111
5. Схема подключения	112
6. Образец кода	115
7. Результат теста	116
 Проект 12: Датчик температуры LM35	
1. Введение.....	117
2. Принцип работы	117
3. Необходимое оборудование	118
4. Схема подключения	119
5. Образец кода	121
6. Результат теста	122
 Проект 13: Переключатель наклона	
1. Введение.....	124
2. Принцип работы	124
3. Необходимое оборудование	125
4. Схема подключения	125
5. Образец кода	127

6. Результат теста	128	
 Проект 14: ИК-пульт дистанционного управления		129
1. Введение.....	129	
2. Принцип работы	130	
3.Штырьки и проводка для инфракрасного приемника	130	
4. Требуемое оборудование	131	
5. Схема подключения	132	
6. Принцип эксперимента	135	
7. Особенности	135	
8. Протокол, как показано ниже	135	
9. Образец кода	138	
10. Результат теста	144	
 Проект 15: Считывание аналогового значения		145
1. Введение.....	146	
2. Требуемое оборудование	146	
3. Характеристики потенциометра	146	
4. Схема подключения	147	
5. Образец кода	149	
6. Результат теста	151	
 Проект 16: микросхема 74HC595		152
1. Введение.....	152	
2. Требуемое оборудование	153	

3. Описание контактов :	154
4. Схема подключения	156
5. Образец кода	158
6. Результат теста	160
 Проект 17: 1-значный сегментный светодиодный дисплей 161	
1. Введение.....	161
2. Требуемое оборудование	161
3. Принцип отображения одноразрядного светодиодного сегментного дисплея	162
4. Схема подключения	163
5. Образец кода	166
6. Результат теста	173
 Проект 18: 4-значный сегментный светодиодный дисплей 174	
1. Введение.....	174
2. Требуемое оборудование	175
3. Принцип отображения 4-значного дисплея	176
4. Схема подключения	177
5. Образец кода	179
6. Результат теста	190
 Проект 19: Светодиодная матрица 8 * 8 191	
1. Введение.....	192
2. Требуемое оборудование	192
3. Принцип отображения точечной матрицы 8 * 8	192

4. Схема подключения	194
5. Пример кода для отображения «0»	196
6. Результат теста	199
Проект 20: 1602 ЖК	200
1. Введение.....	200
2.1602 Основные параметры ЖК-дисплея	200
3. Описание контактов ЖК-дисплея 1602	201
4. Описание интерфейса	201
5.Четыре основные операции для 1602LCD	203
6. Требуемое оборудование	203
7. Подключение	204
8. образец кода A	206
9. образец кода В	216
10. Результат теста	218
Проект 21: Сервоуправление	220
1. Введение.....	220
2. Принцип работы	220
3. Необходимое оборудование	222
4. Способ 1	223
5. Способ 2	228
6. Результат теста	229
Проект 22: Шаговый двигатель	230

1. Введение.....	230
2. Требуемое оборудование	231
3. Особенности	232
4. Параметры шагового двигателя 28BYJ-48	232
5. Схема подключения	233
6. Образец кода	235
7. Результат теста	236
Проект 23: Датчик движения PIR	236
1. Введение.....	236
2. Требуемое оборудование	237
3. Технические характеристики	237
4. Схема подключения	238
5. Образец кода	239
6. Результат теста	241
Проект 24: Аналоговый датчик газа	242
1. Введение.....	242
2. Требуемое оборудование	243
3. Технические характеристики	243
4. Схема подключения	243
5. Образец кода	245
6. Результат теста	246
Проект 25: Трехосное ускорение ADXL345	246

1. Введение.....	246
2. Требуемое оборудование	247
3. Технические характеристики	247
4. Схема подключения	248
5. Образец кода	249
6. Результат теста	254
Проект 26: Ультразвуковой датчик HC-SR04	257
1.Введение Ультразвуковой датчик HC-SR04 - это очень доступный датчик приближения / расстояния, который в основном используется для предотвращения попадания предметов в различные проекты робототехники	257
2. Требуемое оборудование	257
3. Технические характеристики	258
4. Схема подключения	258
5. Образец кода	260
6. Результат теста	262
Проект 27: Модуль джойстика	264
1. Введение.....	264
2. Требуемое оборудование	264
3. Технические характеристики	264
4. Схема подключения	265
5. Образец кода	266
6. Результат теста	267

Проект 28: Модуль реле 5В 269

1. Введение.....	269
2. Требуемое оборудование	269
3. Технические характеристики	270
4. Схема подключения	270
5. Образец кода	271
6. Результат теста	272

Проект 29: Модуль часов DS3231 273

1. Введение.....	273
2. Требуемое оборудование	274
3. Технические характеристики	274
4. Схема подключения	275
5. Образец кода	277
6. Результат теста	279

Проект 30: Датчик температуры и влажности DHT11 281

1. Введение.....	281
2. Требуемое оборудование	282
3. Технические характеристики	283
4. Схема подключения	283
5. Образец кода	284
6. Результат теста	287

Проект 31: Датчик влажности почвы 288

1. Введение.....	288
2. Требуемое оборудование	289
3. Технические характеристики	290
4. Схема подключения	290
5. Образец кода	292
6. Результат теста	293
Проект 32: RFID-модуль RC522	293
1. Введение.....	293
2. Требуемое оборудование	294
3. Технические характеристики	294
4. Схема подключения	295
5. Образец кода	296
6. Результат теста	326
6. Ресурсы	327



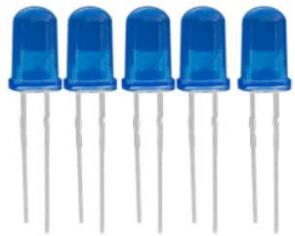
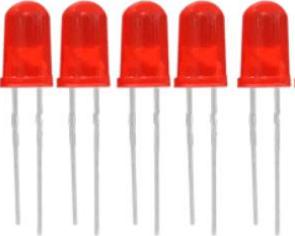
1. Введение

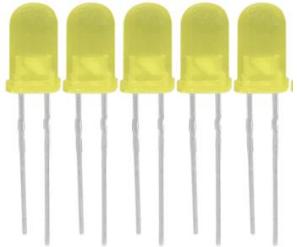
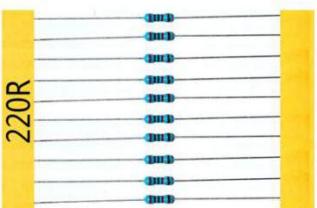
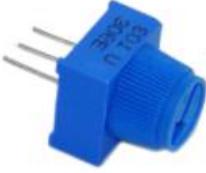
Keyestudio Super Learning Kit подходит для Arduino

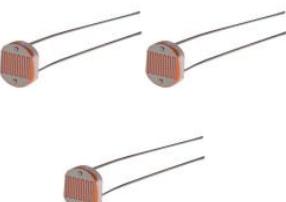
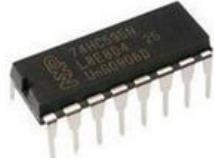
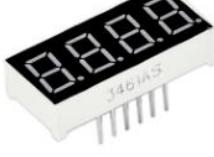
энтузиасты. Этот комплект включает 32 проекта с подробными руководствами, начиная от основ до более сложных проектов. В отличие от других комплектов, он добавляет некоторые функциональные модули, такие как RFID, модули температуры и влажности. Для каждого проекта есть схема подключения и код, что упрощает изучение.

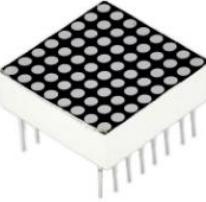
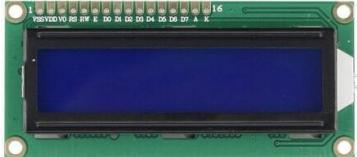
2. СПИСОК КОМПОНЕНТОВ

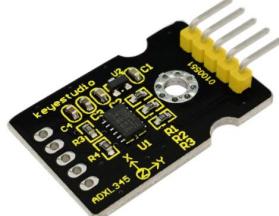
ПРИМЕЧАНИЕ. В комплект KS0077 не входит основная плата; KS0078 В комплект входит плата V4.0; KS0079 KIT Включает плату MEGA 2560.

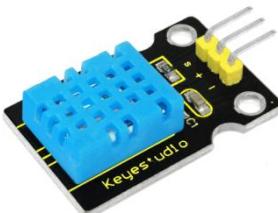
Нет.	наименование товара	QT	Рисунок
1	LED - синий	5	
2	LED - красный	5	

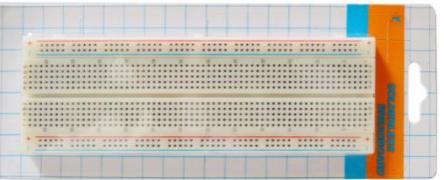
3	Светодиод - желтый	5	
4	LED - RGB	1	
5	220 Ом резистор	8	
6	Резистор 10 кОм	5	
7	Резистор 1 кОм	5	
8	10 кОм Потенциометр	1	
9	Зуммер (активный)	1	
10	Зуммер (пассивный)	1	

11	Большая кнопка Переключатель	4	
12	Датчик наклона шара	2	
13	Фото резистор	3	
14	Датчик пламени	1	
15	Датчик температуры LM35	1	
16	IC 74HC595N 16-контактный DIP	1	
17	7-сегментный светодиод Сегментный дисплей	1	
18	7-сегментный светодиод Сегментный дисплей	1	

19	Светодиодная матрица 8 * 8	1	
20	ЖК-дисплей 2x16	1	
21 год	ИК-приемник	1	
22	ИК-пульт дистанционного управления	1	
23	Серводвигатель	1	
24	Шаговый драйвер	1	
25	Шаговый двигатель	1	

26	Модуль джойстика	1	
27	Модуль реле	1	
28	Датчик движения PIR	1	
29	Аналоговый датчик газа	1	
30	ADXL345 Три оси Модуль ускорения е	1	
31 год	HC-SR04 Ультразвуковой датчик	1	

32	DS3231 Часы Модуль	1	
33	DHT11 Температура и Датчик влажности	1	
34	Влажность почвы Датчик	1	
35 год	RC522 RFID Модуль	1	
36	RFID карта	1	
37	Ключ доступа	1	
38	Заголовки контактов	40	

39	830 лунок Макетная плата	1	
40	Dupont Wire	10	
41 год	Перемычка	30	
42	6-слотовый АА Держатель батареи	1	
43	USB-кабель	1	

3. Установите Arduino IDE и драйвер.

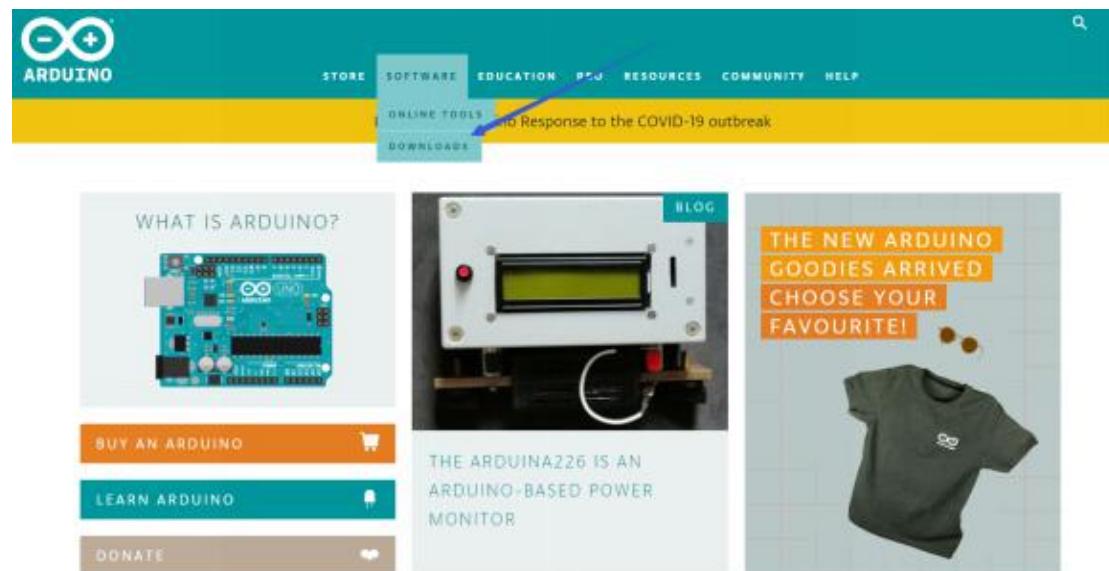
(1) Установка Arduino IDE

Когда мы получим плату управления, нам нужно сначала загрузить Arduino IDE и драйвер.

Вы можете скачать Arduino IDE с официального сайта

<https://www.arduino.cc/> щелкните значок ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ на панели просмотра

нажмите «СКАЧАТЬ», чтобы перейти на страницу загрузки, как показано ниже.



Существуют различные версии IDE для Arduino, просто скачайте версию, совместимую с вашей системой, здесь мы покажем вам, как загрузить и установить версию Arduino IDE для Windows.



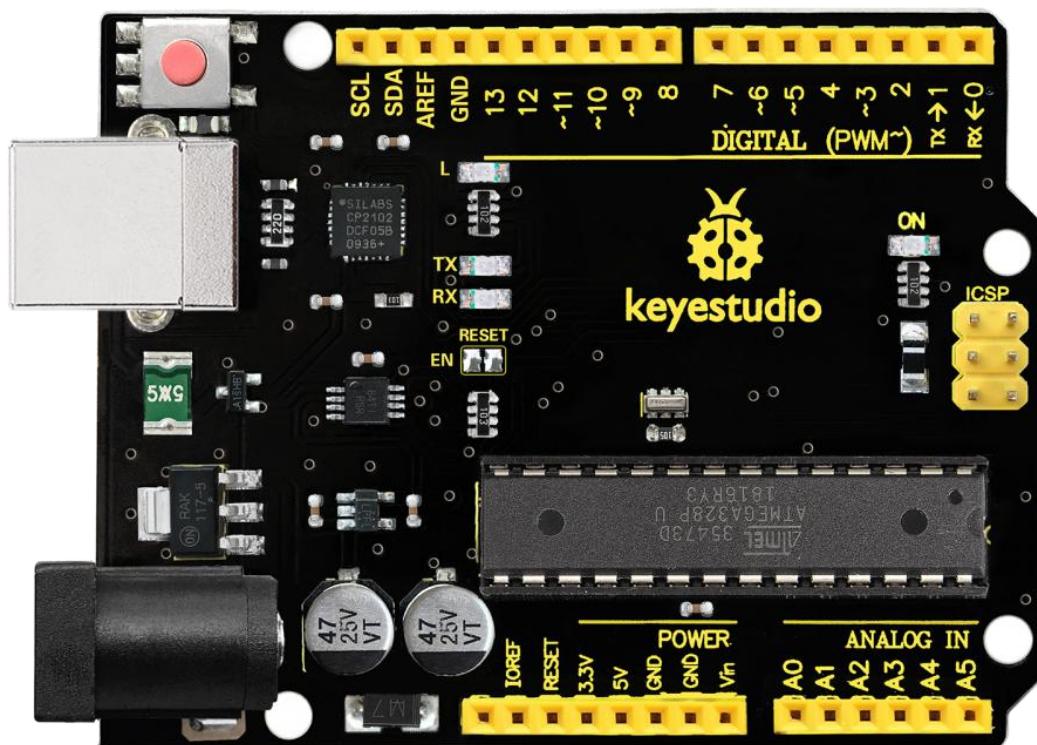
Существует две версии IDE для системы WINDOWS, вы можете выбрать между установщиком (.exe) и пакетами Zip. Мы предлагаем вам использовать первый, который устанавливает непосредственно все, что вам нужно для использования программного обеспечения Arduino (IDE), включая драйверы. С пакетом Zip вам необходимо установить драйверы вручную. Zip-файл также полезен, если вы хотите создать переносную установку.

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). Learn more on how your contribution will be used.

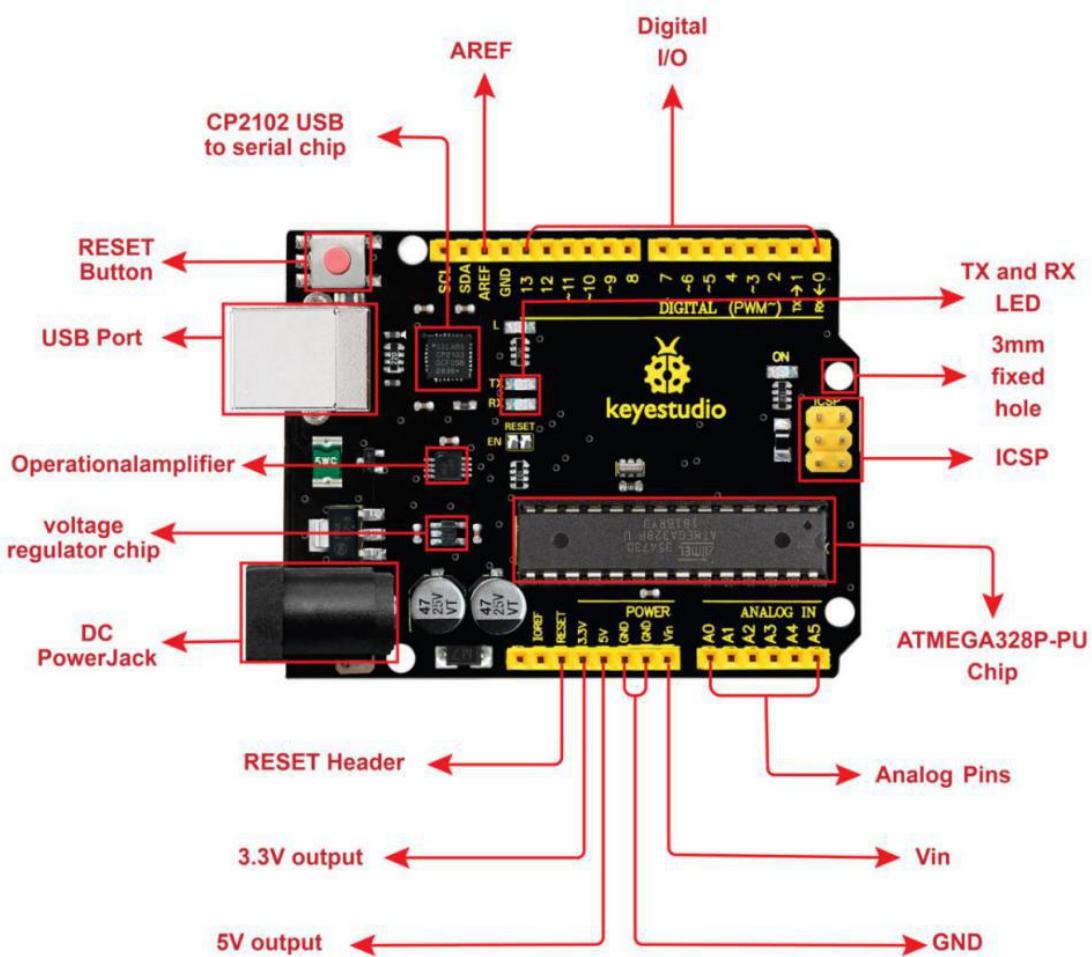
Вам просто нужно нажать ПРОСТО СКАЧАТЬ.

(2) плата разработки keyestudio V4.0

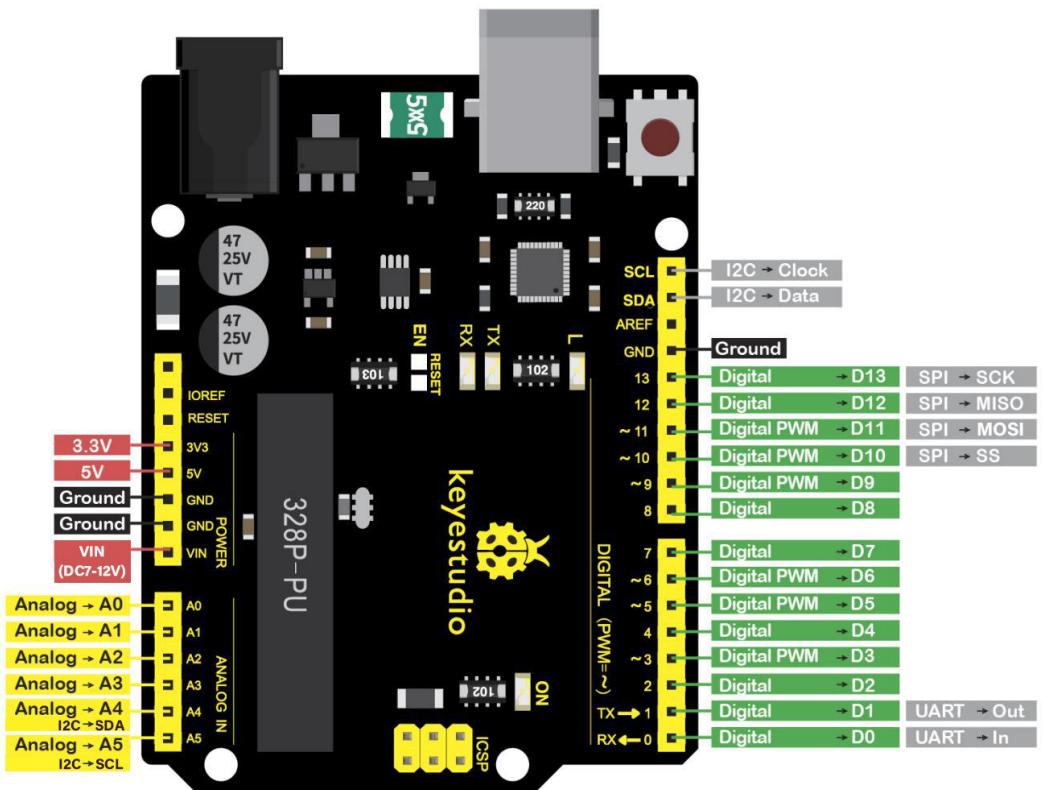
Нам нужно знать отладочную плату keyestudio V4.0, как ядро этого умного автомобиля.



Плата разработки keyestudio V4.0 - это плата Arduino, основанная на микроконтроллере ATmega328P и с микросхемой cp2102 asa UART-to-USB преобразователем.



Он имеет 14 цифровых входов / выходов (из которых 6 могут использоваться как выходы ШИМ), 6 аналоговых входов, кристалл 16 МГц, соединение USB, разъем питания, 2 разъема ICSP и кнопку сброса.



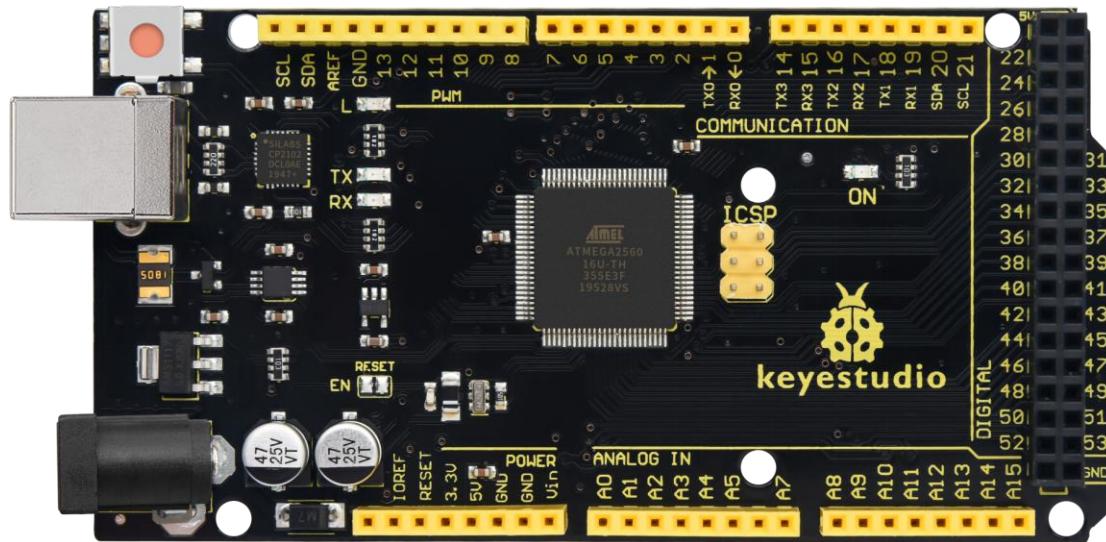
Он содержит все необходимое для поддержки микроконтроллера; для начала просто подключите его к компьютеру с помощью USB-кабеля или подключите его к внешнему разъему питания постоянного тока (7–12 В постоянного тока) или через гнездовые разъемы Vin / GND (7–12 В постоянного тока).

Микроконтроллер	ATmega328P-PU
Рабочее напряжение	5В
Ввод вольтаж (рекомендуемые)	DC7-12V
Цифровые контакты ввода / вывода	14 (D0-D13) (из которых 6 обеспечивают выход ШИМ)
Выводы цифрового ввода / вывода	6 (D3, D5, D6, D9, D10, D11) 6 (A0-A5)
ШИМ Выводы аналогового ввода	
Постоянный ток на выводе ввода / вывода Постоянный ток на выводе 3,3 В	20 мА 50 мА
Флэш-память	32 КБ (ATmega328P-PU), из которых 0,5 КБ используется загрузчиком
SRAM	2 КБ (ATmega328P-PU)
EEPROM	1 КБ (ATmega328P-PU)
Тактовая частота	16 МГц
LED_BUILTIN	D13

(3) Плата Keyestudio MEGA 2560

Keyestudio Mega 2560 R3 - это плата микроконтроллера на базе

ATMEGA2560-16AU, полностью совместимая с ARDUINO MEGA 2560 R3.



Он имеет 54 цифровых входа / выхода (из которых 15 могут использоваться как выходы ШИМ),

16 аналоговых входов, 4 UART (аппаратные последовательные порты), квадцевый генератор 16

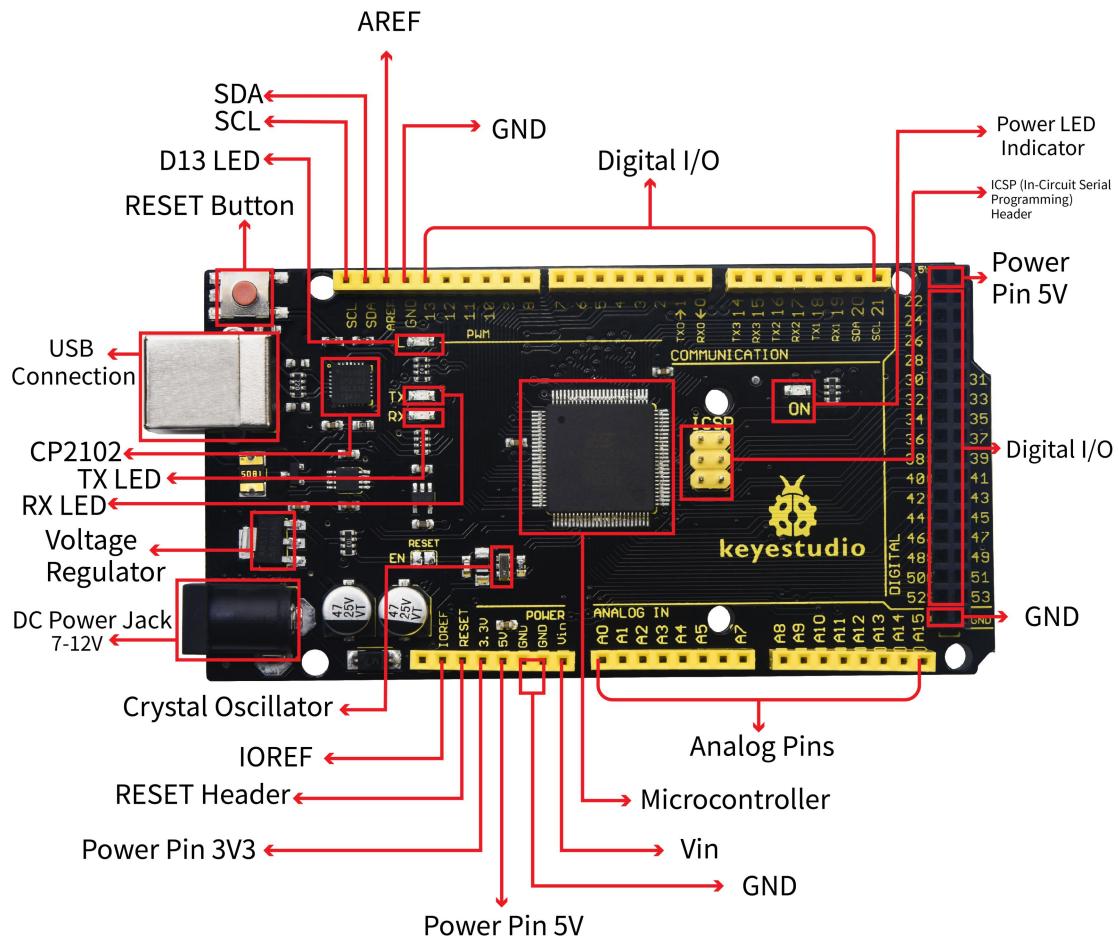
МГц, соединение USB, разъем питания, 1 разъем ICSP, и кнопка сброса. Встроенный порт ICSP

может напрямую записывать прошивку для ATMEGA2560-16AU. Этот чип прошивает прошивку

задолго до того, как покинуть завод, поэтому мы практически не используем его. Мы можем

включить питание через USB-провод, головку постоянного тока и контакты Vin GND. Для

облегчения электромонтажа вам предоставляется провод USB длиной 0,5 м.



Специализированные функции некоторых контактов:

1. Последовательная связь: D0 (RX0) и D1 (TX1); Серийный номер 1: D19 (RX1) и D18 (TX1);

Серийный номер 2: D17 (RX2) и D16 (TX2); Серийный номер 3: D15 (RX3) и D14 (TX3).

Используется для приема (RX) и передачи (TX) последовательных данных TTL. Контакты 0 и 1

также подключены к соответствующим контактам микросхемы последовательного интерфейса

USB-to-TTL CP2102.

2. Контакты PWM (широко-импульсная модуляция): D2 - D13 и D44 - D46.

Обеспечивает 8-битный выход PWM с
функция `analogWrite ()`.

3. Внешние прерывания: D2 (прерывание 0), D3 (прерывание 1), D18 (прерывание 5),
D19 (прерывание 4), D20 (прерывание 3) и D21 (прерывание 2). Эти выводы могут
быть настроены на запуск прерывания на низком уровне, нарастающем или
спадающем фронте или изменении уровня. Подробнее см. Функцию `attachInterrupt ()`.

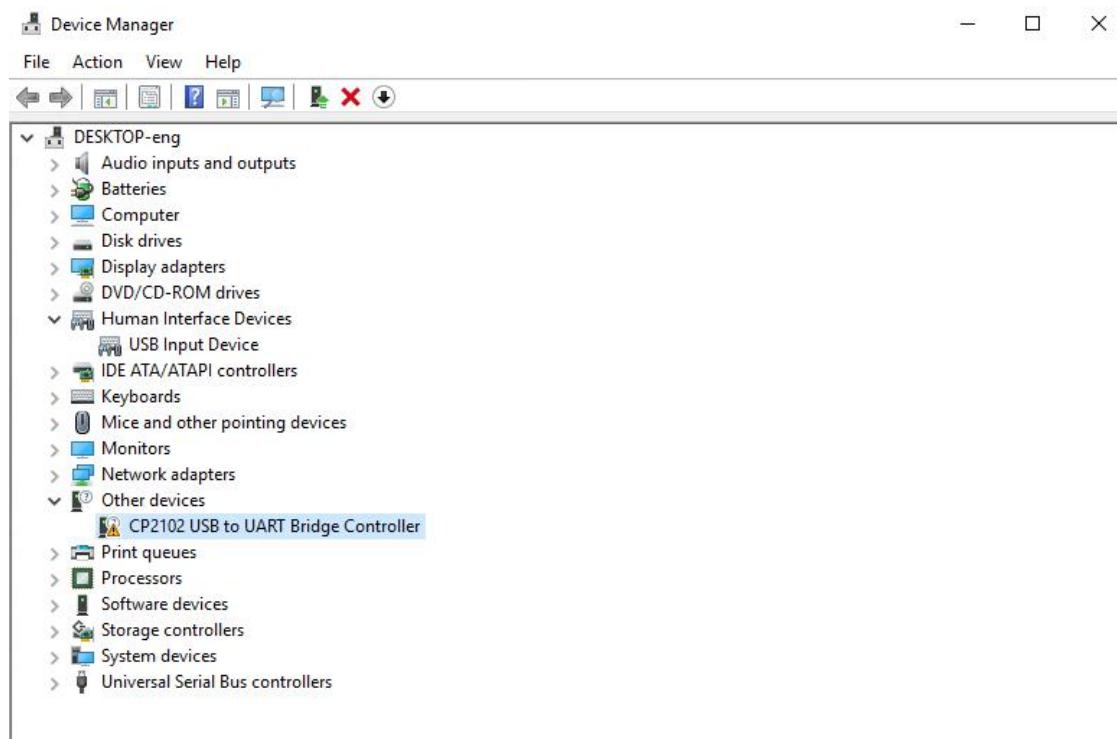
4. Связь SPI: D53 (SS), D52 (SCK), D51 (MOSI), D50 (MISO). Эти контакты
поддерживают связь SPI с использованием библиотеки SPI. Контакты SPI
также разбиты на заголовке ICSP, который физически совместим с Arduino
Uno.

5. Связь IIC: D20 (SDA); D21 (SCL). Поддержите связь TWI с помощью
библиотеки Wire.

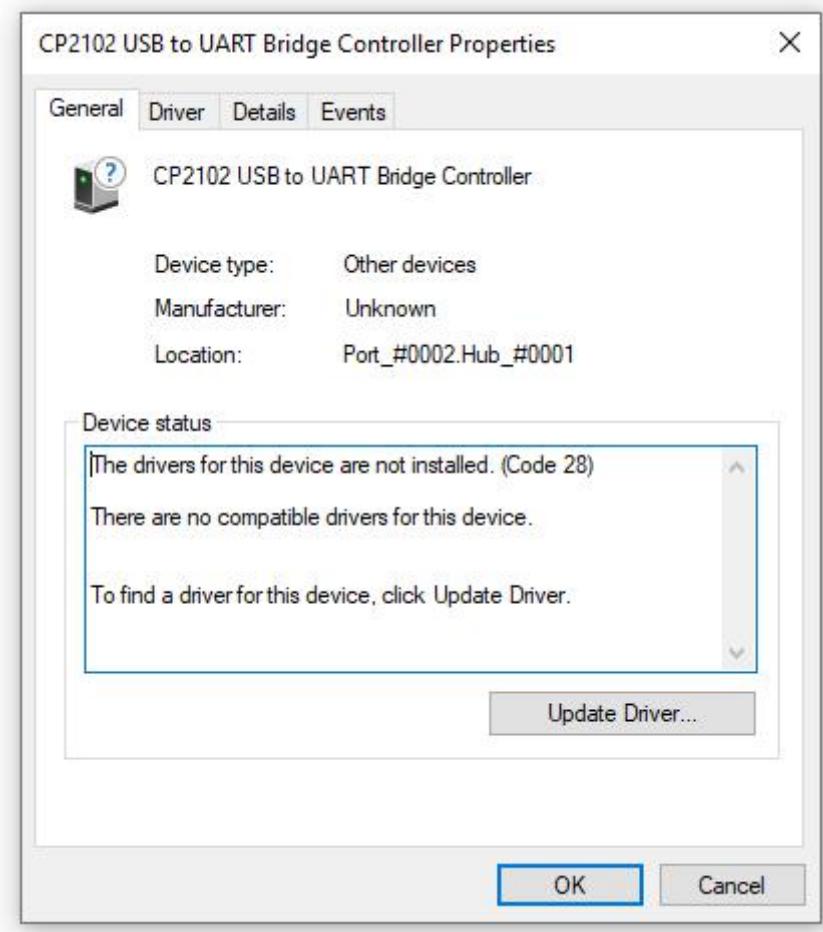
(4) Установка драйвера платы V4.0

Установим драйвер платы keyestudio V4.0. Микросхема USB-TTL на плате V4.0
использует последовательный чип CP2102. Программа драйвера этого чипа
включена в версию Arduino 1.8 и выше, что удобно. Подключите USB-порт
платы, компьютер может распознать оборудование и автоматически
установить драйвер CP2102.

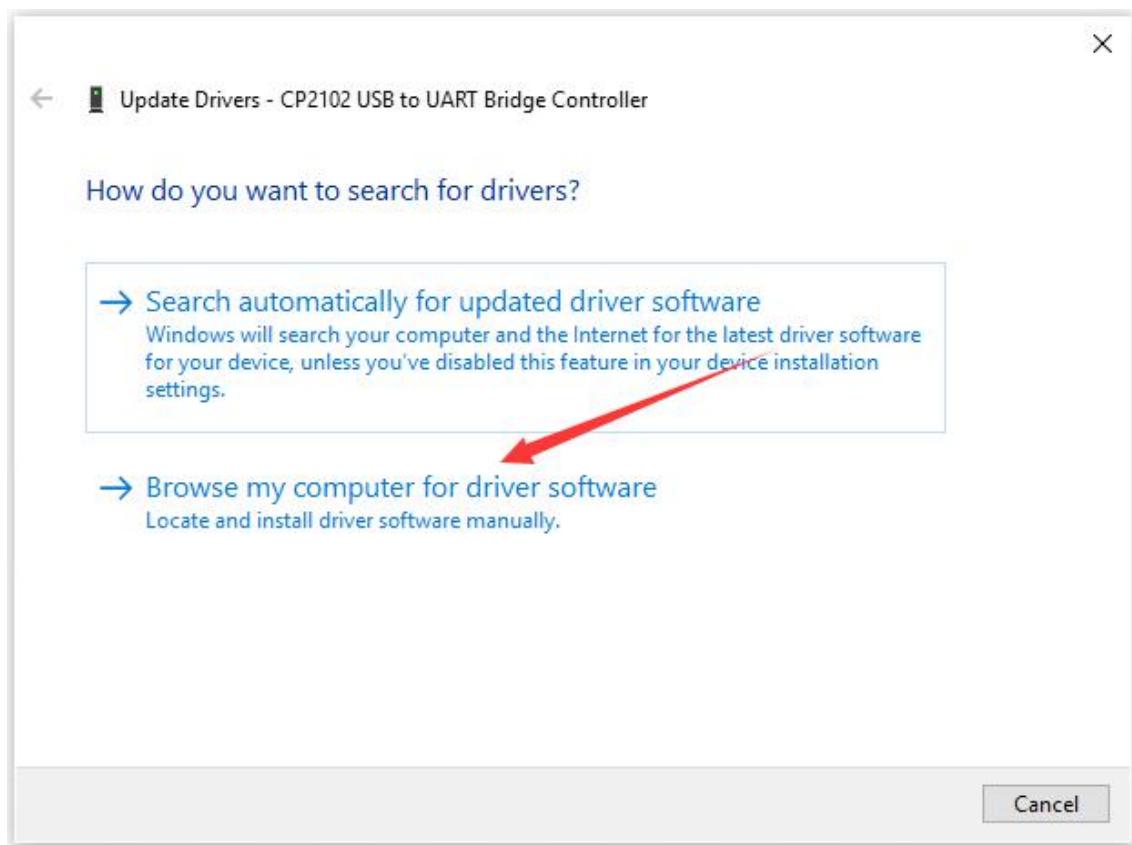
Если установка завершилась неудачно или вы собираетесь установить вручную, откройте диспетчер устройств компьютера. Щелкните правой кнопкой мыши Компьютер ----- Свойства ----- Диспетчер устройств



На странице стоит желтый восклицательный знак, что означает неудачную установку. Затем мы дважды щелкаем оборудование и обновляем драйвер.



Нажмите «OK», чтобы перейти на следующую страницу, нажмите «Найдите на моем компьютере обновленное программное обеспечение драйвера», найдите установленное или загруженное программное обеспечение ARDUINO. Как показано ниже



Существует [ВОДИТЕЛИ](#) папка в [Программное обеспечение Arduino установлено](#)

пакет ( arduino-1.8.12) , откройте папку с драйверами и вы

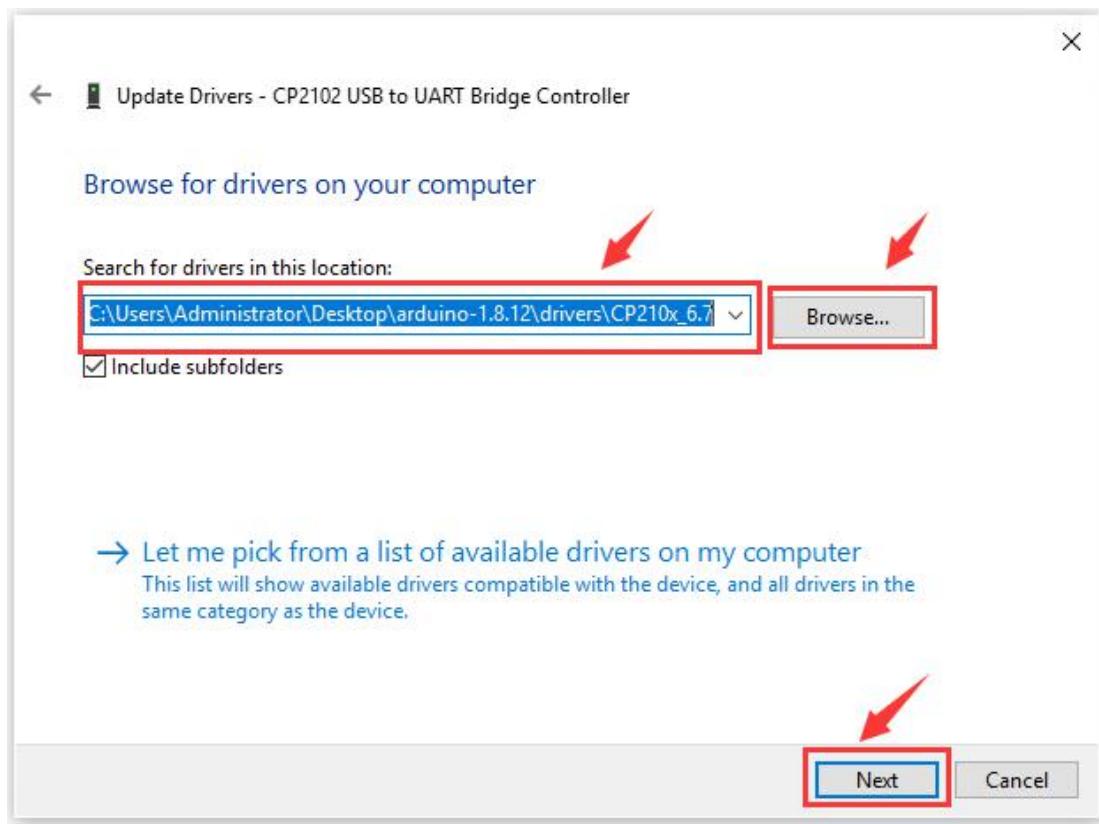
можно увидеть водителя [Чипы серии CP210X](#) .

Нажимаем «Обзор», затем узнаем [Водитель](#) папку, или вы можете ввести «драйвер»

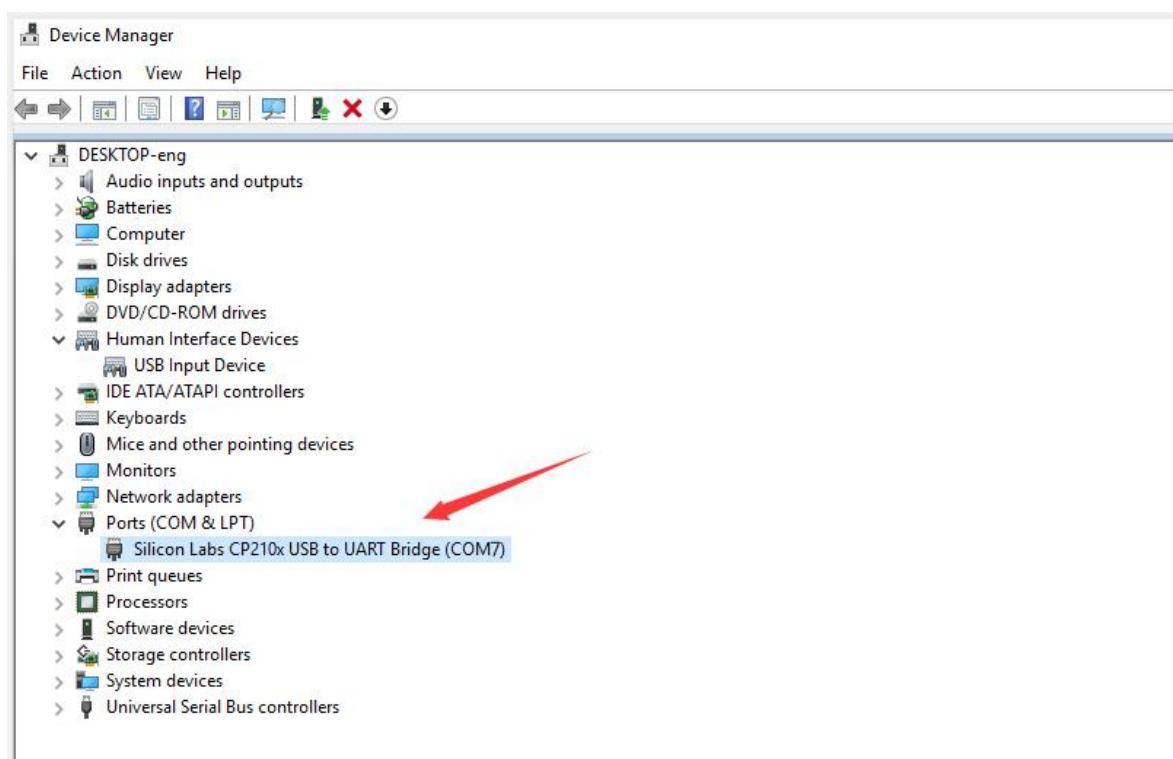
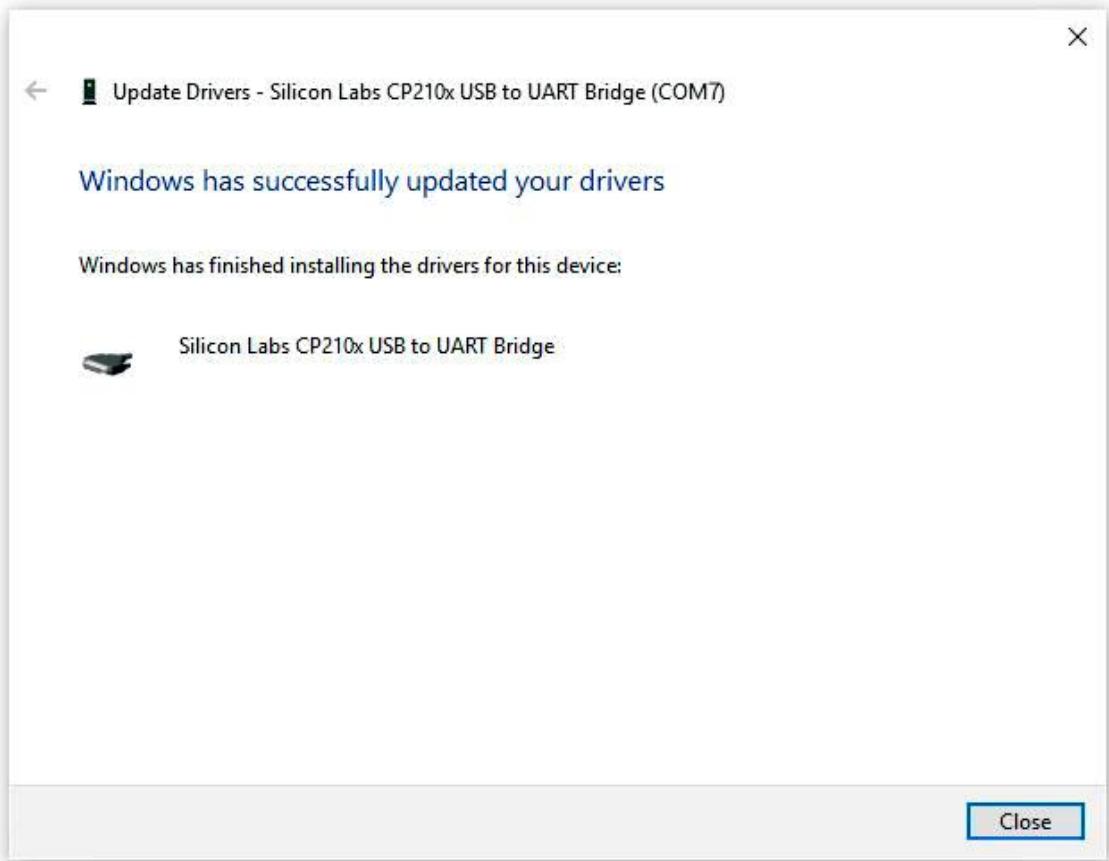
для поиска в прямоугольном поле, а затем нажать «Далее», драйвер будет успешно

установлен. (Я помещаю папку с программным обеспечением Arduino на рабочий стол,

вы можете пойти по моему пути)

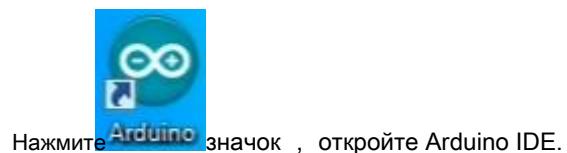


Откройте диспетчер устройств, и мы обнаружим, что желтый восклицательный знак исчезнет. Драйвер CP2102 успешно установлен.

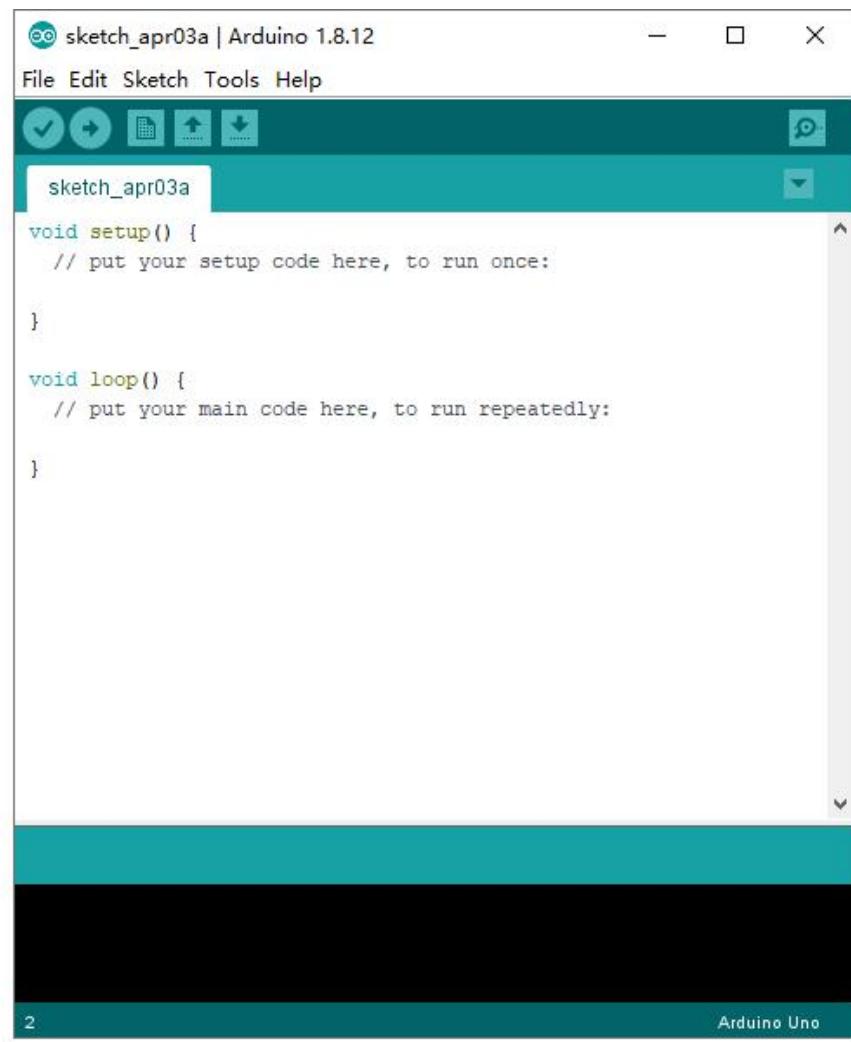


Метод установки платы keyestudio MEGA 2560 и платы V4.0 одинаков.

(5) Настройка Arduino IDE



Нажмите значок , откроите Arduino IDE.

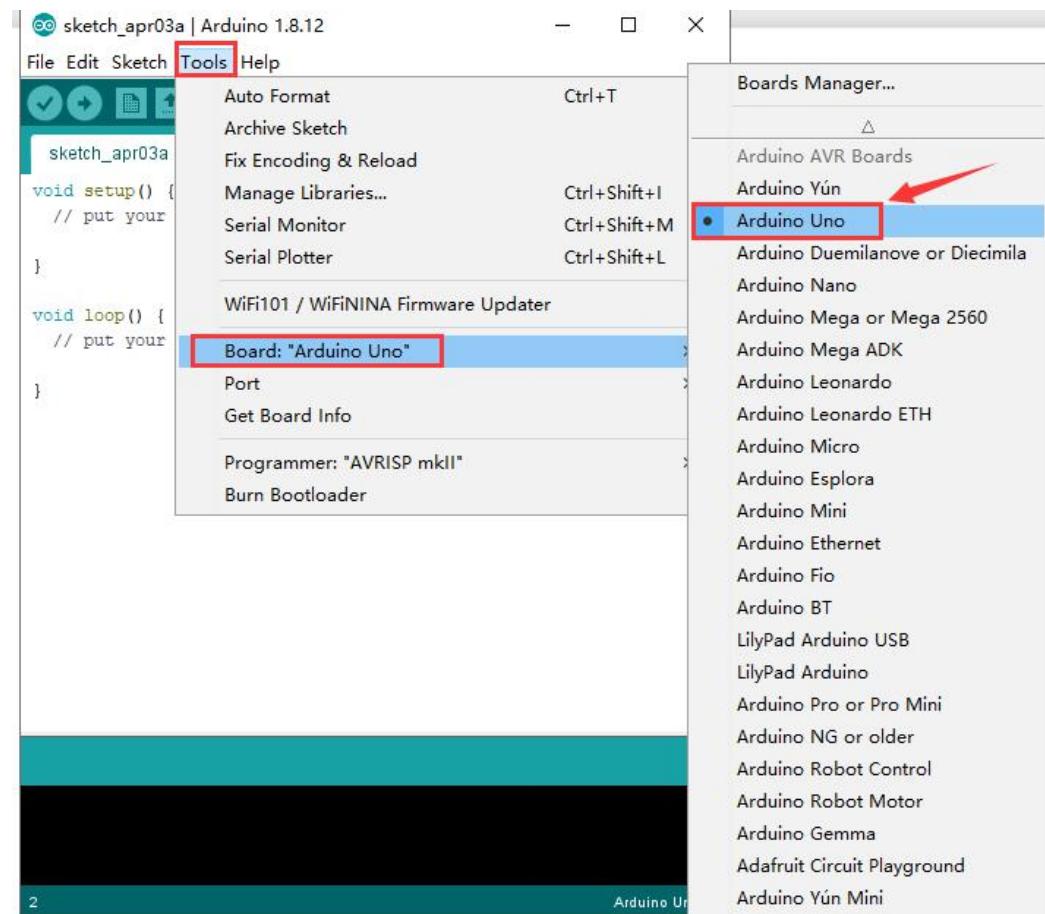


```
void setup() {  
    // put your setup code here, to run once:  
  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
  
}
```

Чтобы избежать ошибок при загрузке программы на плату, вам необходимо выбрать правильную плату Arduino, которая соответствует плате, подключенной к вашему компьютеру.

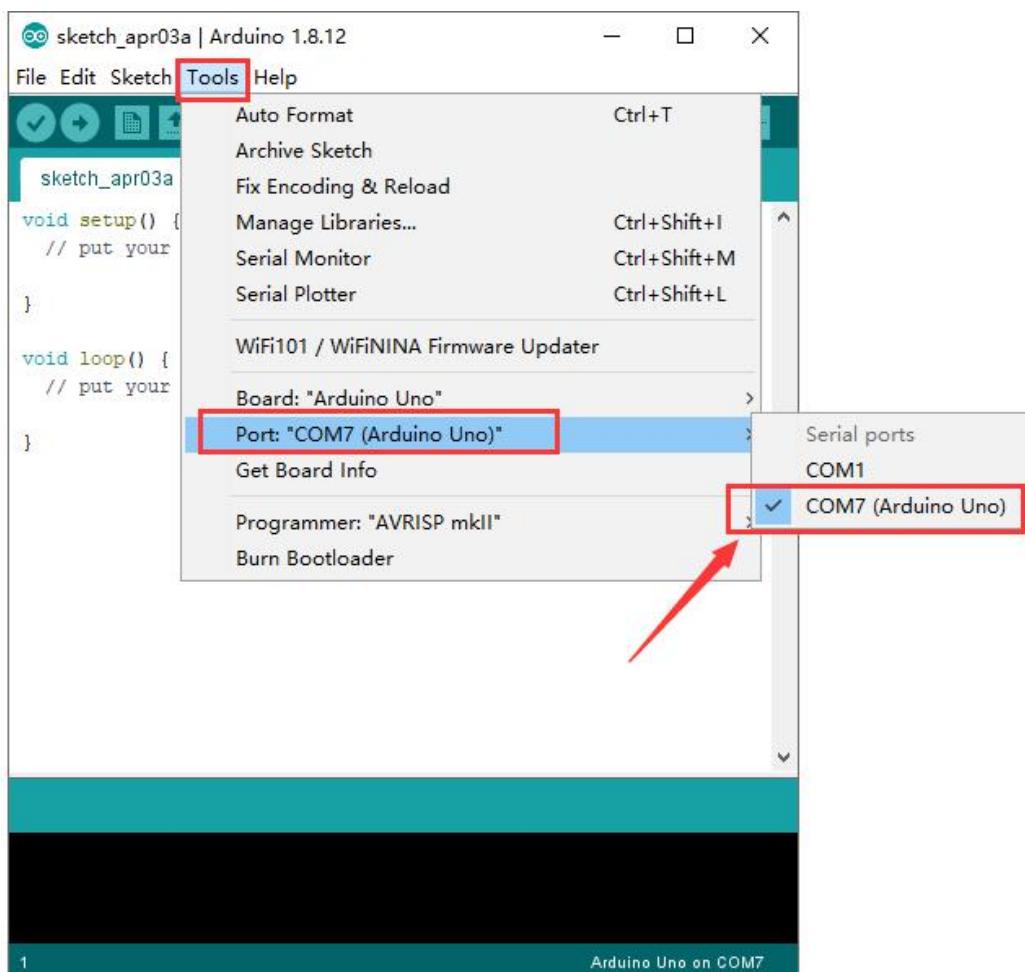
Затем вернитесь в программное обеспечение Arduino, вам нужно нажать Инструменты →

Плата, выбрать плату. (как показано ниже)

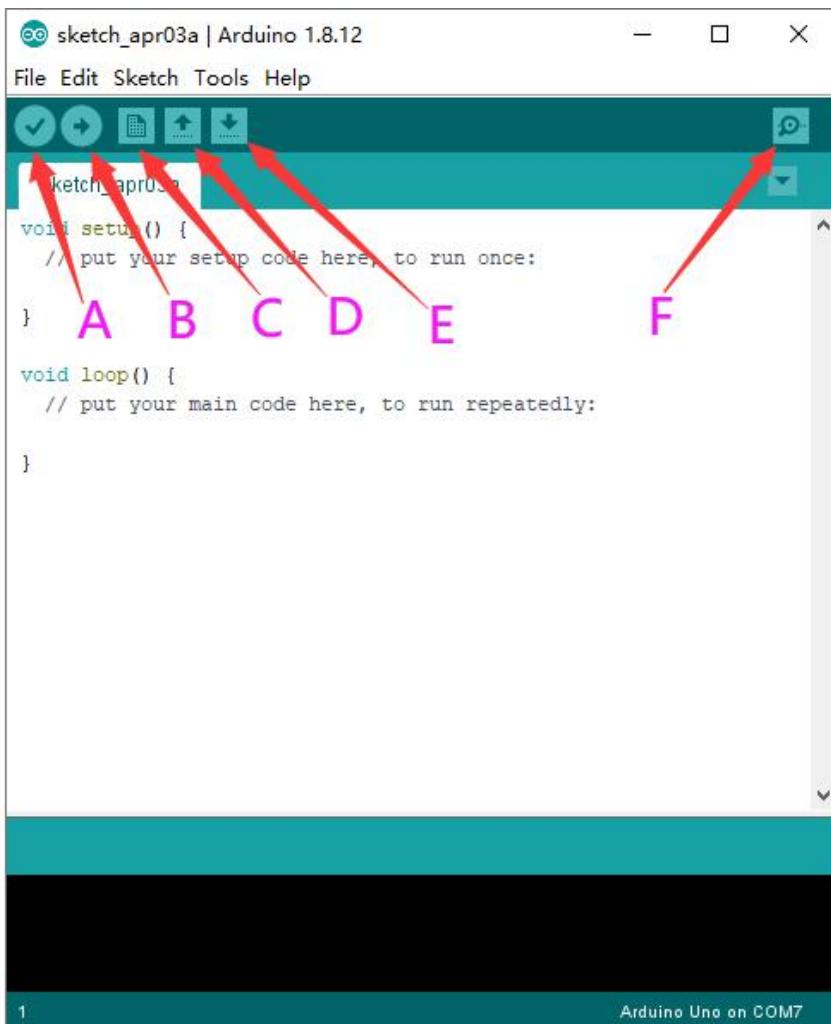


Затем выберите правильный СОМ-порт (вы увидите соответствующий

СОМ-порт после успешной установки драйвера)



Прежде чем загружать программу на плату, давайте продемонстрируем функцию каждого символа на панели инструментов Arduino IDE.



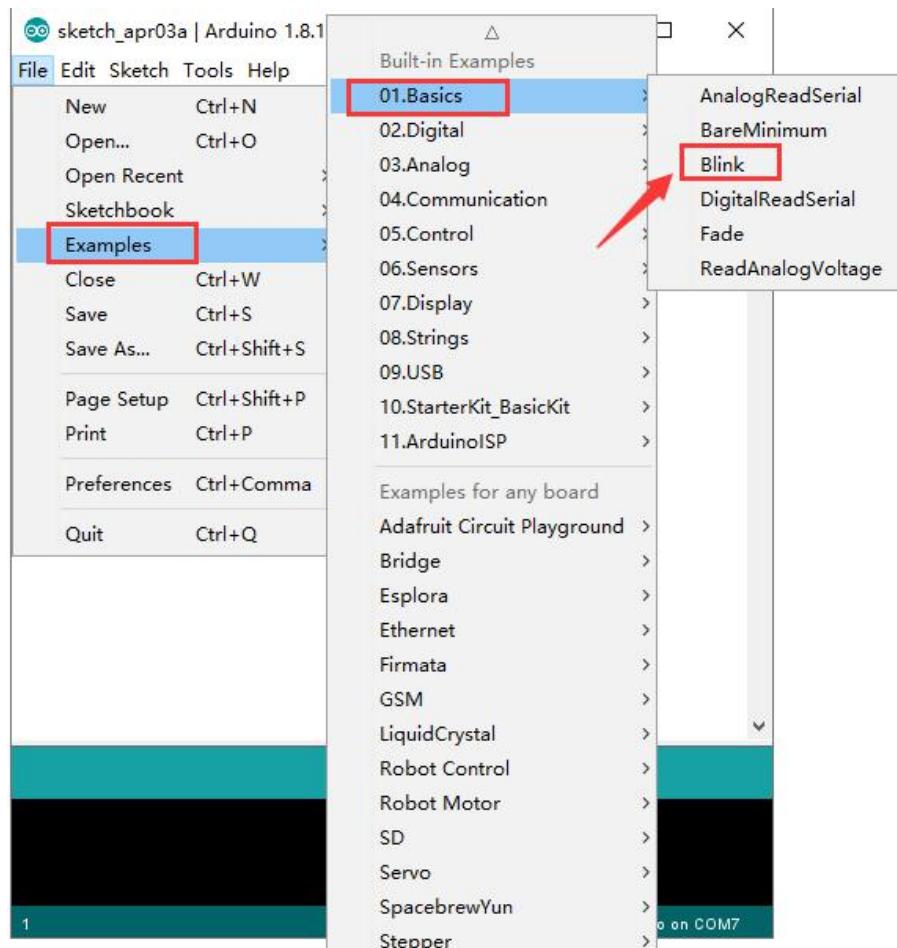
А- Используется для проверки наличия ошибок компиляции.

Б- Используется для загрузки эскиза на плату Arduino. С- Используется для создания окна быстрого доступа к новому эскизу. Д- Используется для непосредственного открытия примера эскиза. Е- Используется для сохранения эскиза.

Ф- Используется для отправки последовательных данных, полученных от платы, на последовательный монитор.

(6) Запустить первую программу

Откройте файл, чтобы выбрать пример , выберите Мигать из БАЗОВЫЙ , как показано ниже



```
This example code is in the public domain.  
http://www.arduino.cc/en/Tutorial/Blink  
//  
  
// the setup function runs once when you press reset or power the  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is t  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making  
  delay(1000); // wait for a second  
}
```

Установите плату и СОМ-порт, соответствующая плата и СОМ-порт показаны в правом нижнем углу IDE.

```
This example code is in the public domain.  
http://www.arduino.cc/en/Tutorial/Blink  
//  
  
// the setup function runs once when you press reset or power the  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is t  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making  
  delay(1000); // wait for a second  
}
```



Нажмите чтобы начать компилировать программу, проверьте ошибки.

```
Blink | Arduino 1.8.12
File Edit Sketch Tools Help

Blink
This example code is in the public domain.

http://www.arduino.cc/en/Tutorial/Blink

// the setup function runs once when you press reset or power the
void setup() {
    // initialize digital pin LED_BUILTIN as an output.
    pinMode(LED_BUILTIN, OUTPUT);
}

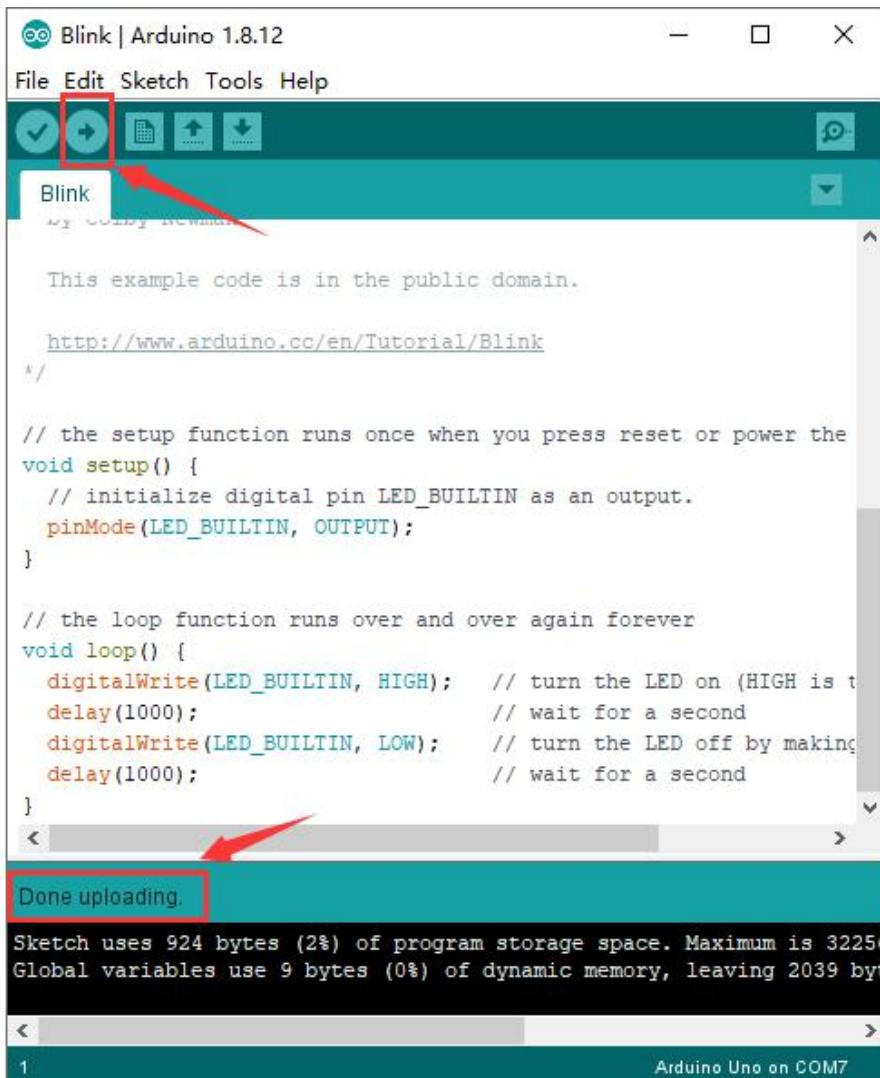
// the loop function runs over and over again forever
void loop() {
    digitalWrite(LED_BUILTIN, HIGH);      // turn the LED on (HIGH is t
    delay(1000);                      // wait for a second
    digitalWrite(LED_BUILTIN, LOW);       // turn the LED off by making
    delay(1000);                      // wait for a second
}

Done compiling.
Sketch uses 924 bytes (2%) of program storage space. Maximum is 32
Global variables use 9 bytes (0%) of dynamic memory, leaving 2039
1
Arduino Uno on COM7

```



Нажмите чтобы загрузить программу, загрузите успешно.



Загрузите программу успешно, встроенный светодиод горит на 1 с, гаснет на 1 с. Поздравляю, вы закончили первую программу.

Если это плата keyestudio MEGA 2560, выберите плату Arduino MEGA или
MEGA 2560.

4. Как добавить библиотеку?

Что такое библиотеки?

Библиотеки - это набор кода, который упрощает подключение к датчику, дисплею, модулю и т. Д.

Например, встроенная библиотека LiquidCrystal помогает общаться с ЖК-дисплеями. В Интернете для загрузки доступны сотни дополнительных библиотек.

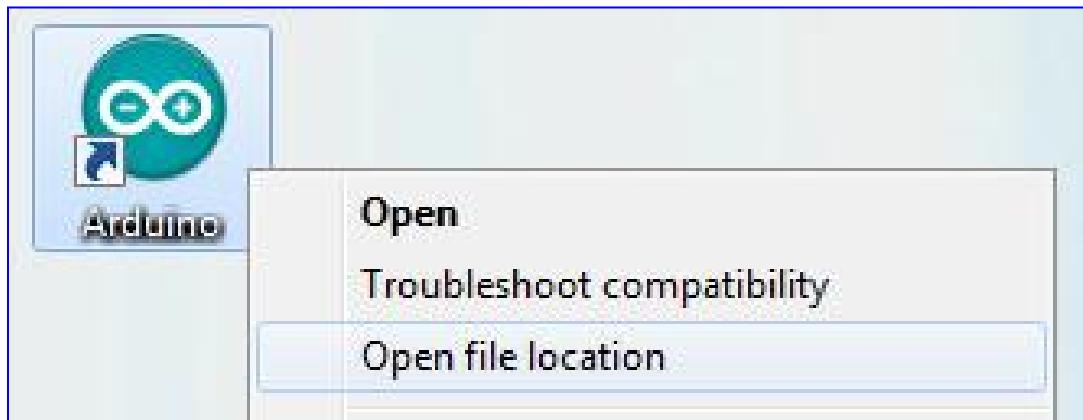
Встроенные библиотеки и некоторые из этих дополнительных библиотек перечислены в справочнике.

Как установить библиотеку?

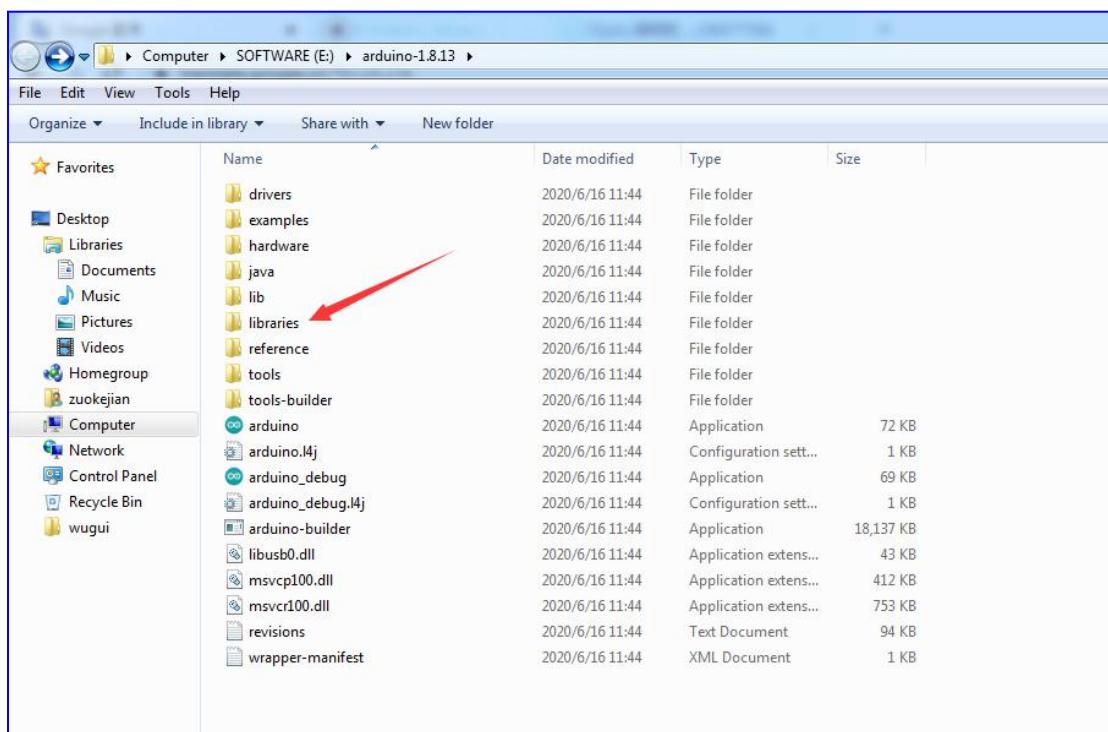
Здесь мы познакомим вас с самым простым способом добавления библиотек.

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

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



Шаг 2 Войдите в нее, чтобы найти папку с библиотеками, эта папка является файлом библиотеки Arduino.



Шаг 3 Далее, чтобы узнать Папка "библиотеки" этого набора (видно по ссылке <https://fs.keyestudio.com/>)

👤 > 1.Arduino > Starter kit >

KS0077 (78, 79) keyestudio Super Learning Kit for ARDU...

Click here to describe this folder and turn it into a Space [Show examples](#)

★ Pin or drag files and folders here for quick access

4 folders

Create ⌂ ⌂ ⌂ ⌂ ⌂

Name	Modified	Recent activity	Type
▶ 1. about keyestudio	5/21/20, 4:45 pm	--	File folder
▶ 2. Arduino software	5/21/20, 4:45 pm	--	File folder
▶ 3. Getting started with ard...	5/21/20, 4:45 pm	--	File folder
▶ 4. Tutorial	6/10/20, 2:13 pm	--	File folder

👤 > ... > Starter kit > KS0077 (78, 79) keyestudio Super Learning Kit for ARDUINO >

4. Tutorial

Click here to describe this folder and turn it into a Space [Show examples](#)

★ Pin or drag files and folders here for quick access

2 folders, 1 file

Create ⌂ ⌂ ⌂ ⌂ ⌂

Name	Modified	Recent activity	Type
▶ libraries	5/21/20, 4:45 pm	--	File folder
▶ project code	5/21/20, 4:45 pm	--	File folder
▶ KS0077 (78,79) Super Lear...	5/27/19, 3:40 pm	✖ You moved 7/22/20, 3:43 pm	WPS PDF 文档

libraries

Click here to describe this folder and turn it into a Space [Show examples](#)

★ Pin or drag files and folders here for quick access

7 folders

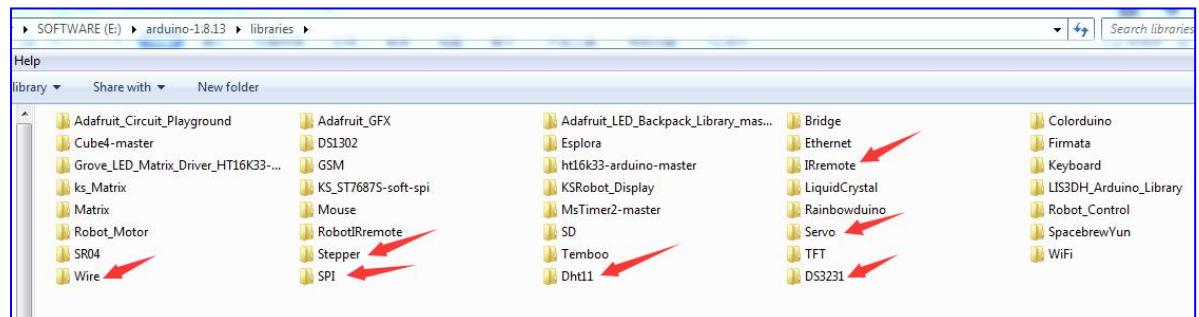


Name	Modified	Recent activity	Type
Dht11	5/21/20, 4:45 pm	--	File folder
DS3231	5/21/20, 4:45 pm	--	File folder
IRremote	5/21/20, 4:45 pm	--	File folder
Servo	5/21/20, 4:45 pm	--	File folder
SPI	5/21/20, 4:45 pm	--	File folder
Stepper	5/21/20, 4:45 pm	--	File folder
Wire	5/21/20, 4:45 pm	--	File folder

Вам просто нужно скопировать и вставить вышеуказанные библиотеки в

папку библиотек Arduino IDE .

Затем библиотеки этого набора устанавливаются успешно, как показано ниже.



Обратите внимание, что загрузка программного обеспечения Arduino и установка драйвера для платы keyestudio Mega 2560 R3 аналогичны плате Arduino V4.0.

5. детали проекта

Проект 1: Привет, мир



Hello world

1. Введение

Что касается начала, то начнем с простого. В этом проекте вам понадобятся только Arduino и USB-кабель, чтобы запустить «Hello World!». эксперимент. Это не только тест связи между вашим Arduino и ПК, но и учебный проект для вас, чтобы вы впервые попробовали в мире Arduino!

2. Требуется оборудование

- Плата V4.0 или плата MEGA 2650 * 1
- USB-кабель * 1

3. образец кода

После установки драйвера для Arduino давайте откроем программное обеспечение Arduino и скомпилируем код, который позволяет Arduino печатать «Hello World!». по вашему указанию. Конечно, вы можете скомпилировать код для Arduino, который будет постоянно повторять «Hello World!». без инструкции.

Просто **Оператор if ()** сделает трюк с инструкциями. когда Arduino получает инструкцию, а затем печатает «Hello World!».

```
/*  
супер обучающий комплект keyestudio Project
```

```
1
```

```
Привет, мир
```

```
http://www.keyestudio.com
```

```
* /
```

```
int val; // определяем переменную val void
```

```
setup ()
```

```
{
```

```
Serial.begin (9600); // устанавливаем скорость передачи 9600. }
```

```
пустой цикл ()
```

```
{
```

```
val = Serial.read (); // считываем инструкцию или символ с ПК на
```

Arduino и назначьте их Val.

```
if (val == 'R') // определяем, является ли полученная инструкция или символ «R».
```

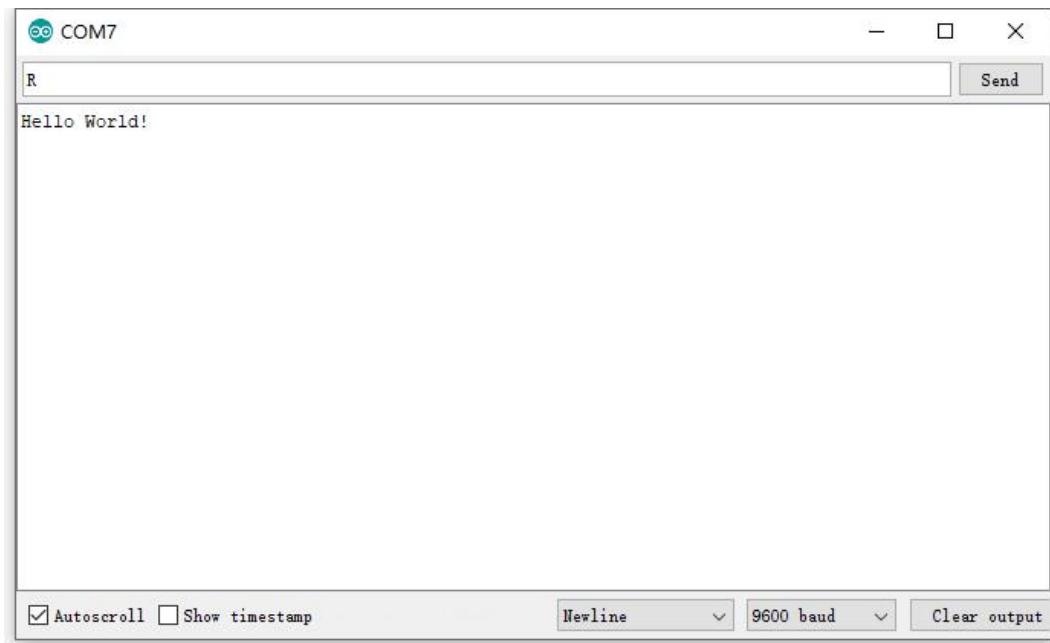
```
{// если это «R»,
```

```
    Serial.println ("Hello World!"); // выводим строку "Hello World ! ". } } //
```

```
||||||||||||||||||||||||||||||||||||||||||
```

4. результат теста

Щелкните, чтобы открыть монитор последовательного порта, введите «R», ПК получит информацию от Arduino Hello World!



После выбора правильного порта эксперимент станет для вас легким!

Проект 2: мигание светодиода



1. Введение

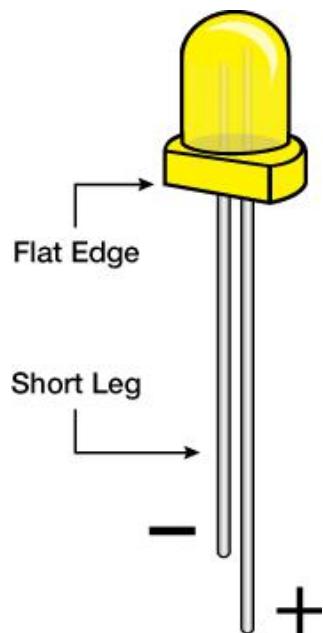
Эксперимент с миганием светодиода довольно прост. В "Hello World!" Программа, нам попался светодиод. На этот раз мы собираемся подключить светодиод к одному из цифровых контактов, а не использовать светодиод 13, припаянный к плате. Помимо Arduino и USB-кабеля, вам понадобятся дополнительные детали, как показано ниже.

2. Требуется оборудование

- Плата V4.0 или плата MEGA 2650 * 1
- Красный светодиод M5 * 1
- Резистор 220 Ом * 1
- Макетная плата * 1
- Перемычка макетной платы * 2
- USB-кабель * 1

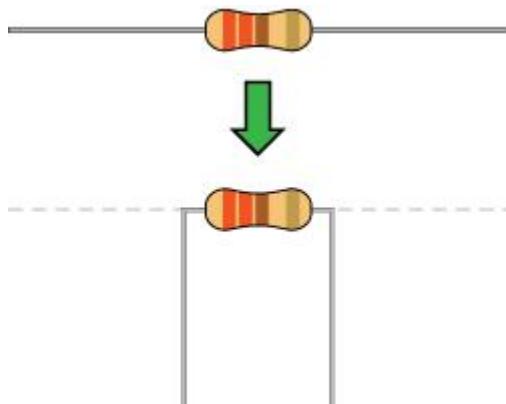
3. Мало знаний

Светодиод - это тип полупроводника, называемый «светоизлучающим диодом», который представляет собой электронное устройство, изготовленное из полупроводниковых материалов (кремний, селен, германий и т. д.). Это дублированный индикатор, цифровой и текстовый дисплей в схеме и устройстве. У него есть положительные и отрицательные полюса. Короткая нога - отрицательный полюс, длинная - положительный полюс.

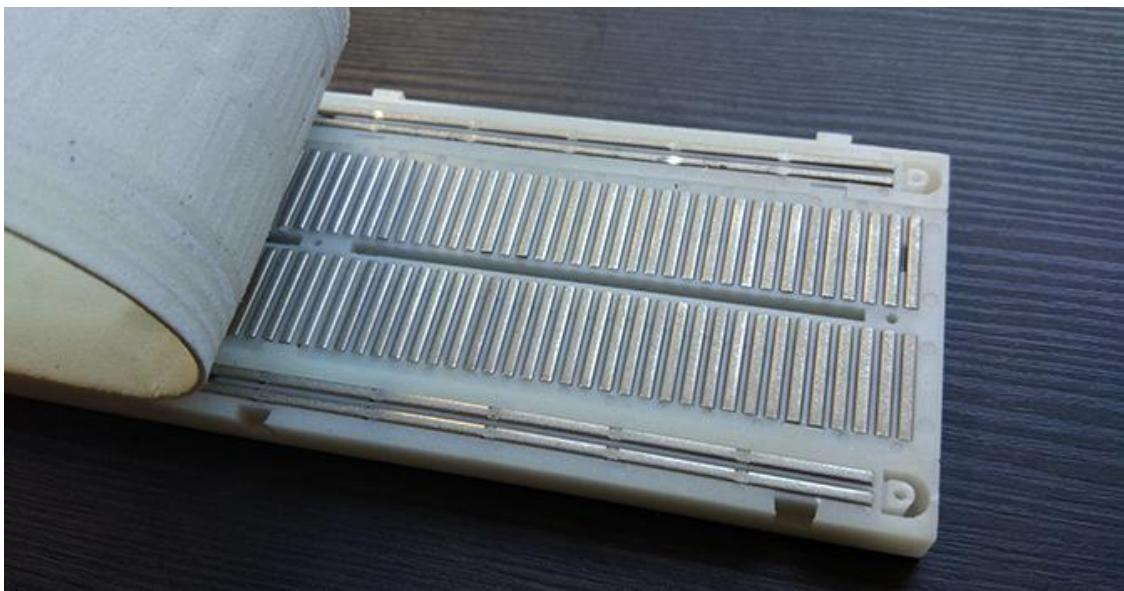
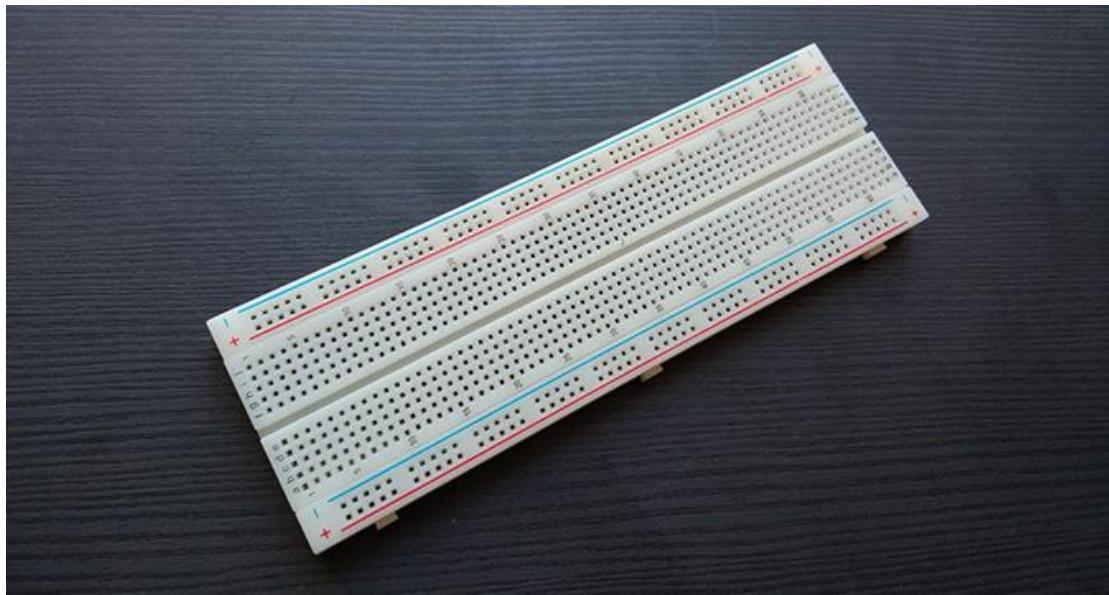


Резистор : Резистор - это электронный компонент в цепи, который ограничивает и регулирует ток. Его единица - (Ω).

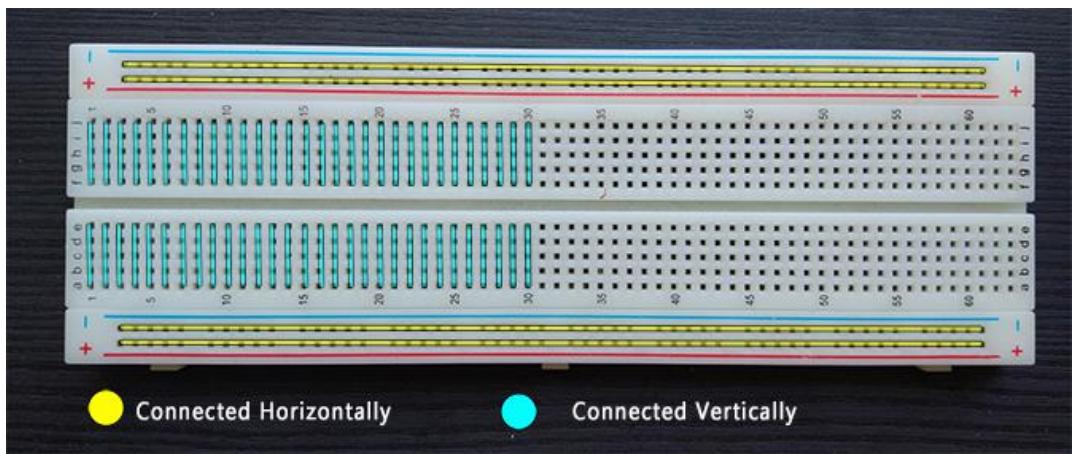
Единицы измерения больше ома - килоом (кОм) и мегаом (МОм). При использовании, помимо размера сопротивления, вы также должны обращать внимание на его мощность. В проекте выводы на обоих концах резистора должны быть согнуты под углом 90 °, чтобы правильно поместиться на макетной плате. Если поводок слишком длинный, его можно обрезать до подходящей длины.



Макетная плата используется для быстрого создания и тестирования схем перед окончательной доработкой схемы. На макетной плате есть множество отверстий, в которые можно вставить такие компоненты схемы, как микросхемы и резисторы. Типичный макет показан ниже:



На макетной плате есть полосы металла, которые проходят под платой и соединяют отверстия в верхней части платы. Металлические полосы расположены, как показано ниже. Обратите внимание, что верхний и нижний ряды отверстий соединяются горизонтально, а остальные отверстия соединяются вертикально.

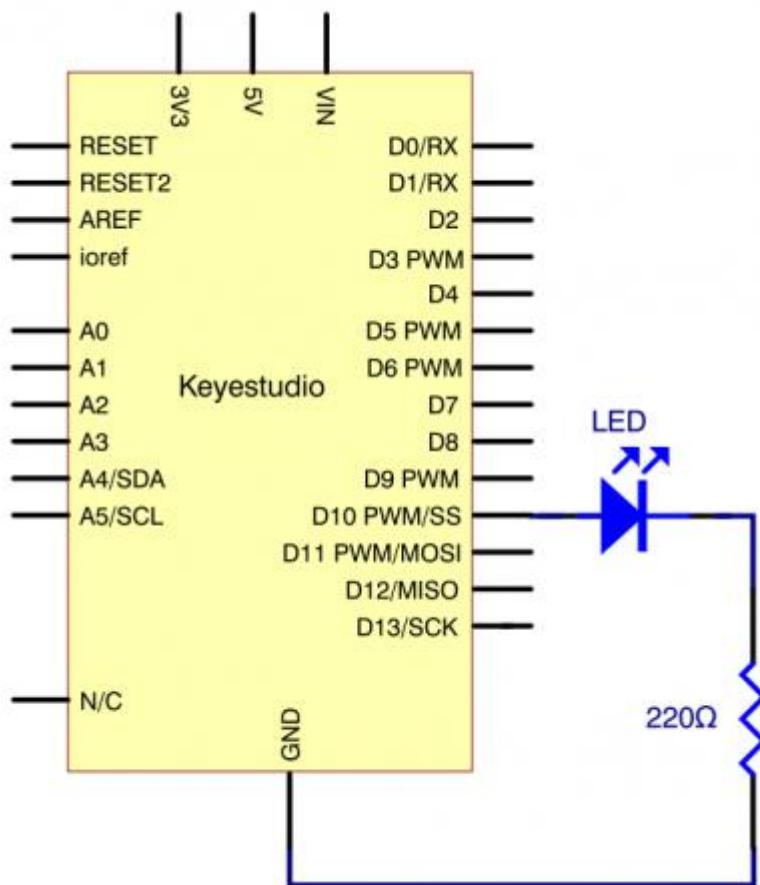


Для использования макетной платы в отверстия вставляются ножки компонентов.

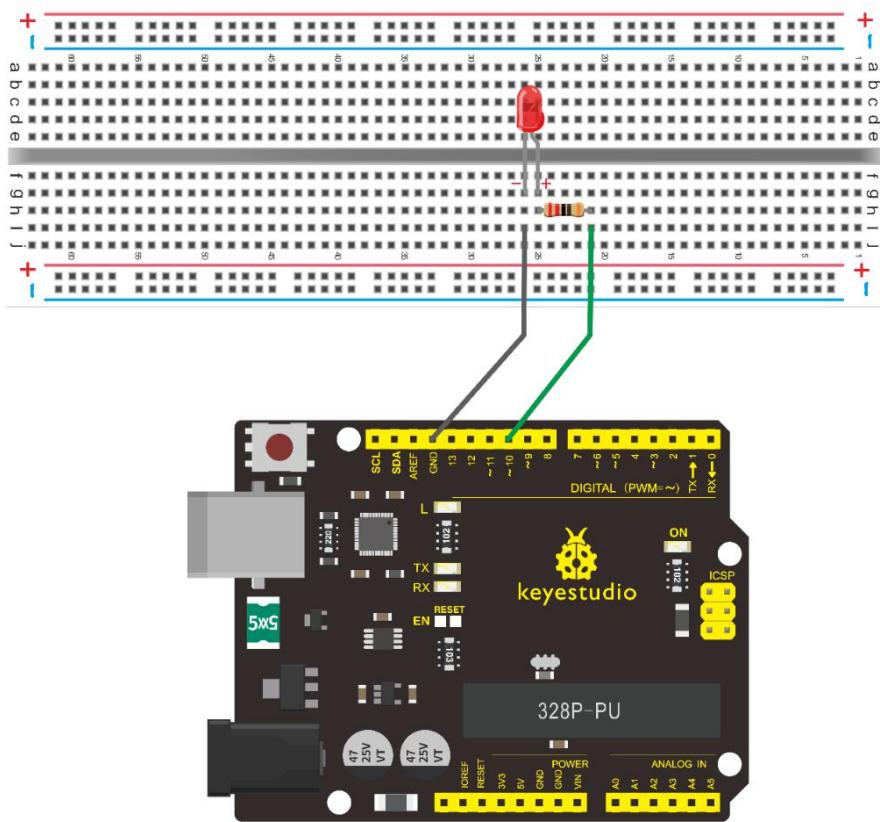
Каждый набор отверстий, соединенных металлической лентой, образует анод.

4. схема подключения

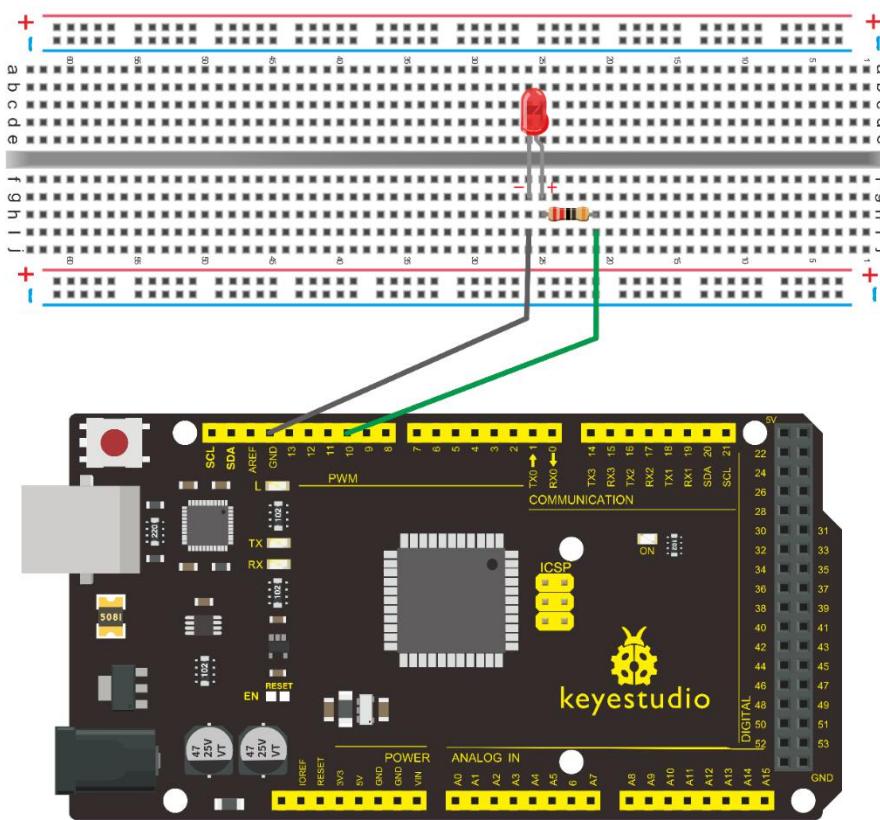
Мы следуем диаграмме ниже из экспериментальной схематической ссылки. Здесь мы используем цифровой контакт 10. Мы подключаем светодиод к резистору 220 Ом, чтобы избежать повреждения светодиода сильным током.



Подключение для V4.0



Подключение для 2560



5. образец кода

```
/*
```

```
супер обучающий комплект keyestudio Project
```

```
2
```

```
Мигать
```

```
http // www.keyestudio.com
```

```
*/
```

```
int ledPin = 10; // определяем цифровой контакт 10. void
```

```
setup ()
```

```
{
```

```
pinMode (ledPin, OUTPUT); // определяем вывод с подключенным светодиодом как выход. }
```

```
пустой цикл ()
```

```
{
```

```
digitalWrite (ledPin, HIGH); // включаем светодиод. задержка
```

```
(1000); // ждем секунду.
```

```
digitalWrite (ledPin, LOW); // выключаем светодиод. задержка
```

```
(1000); // ждем секунду
```

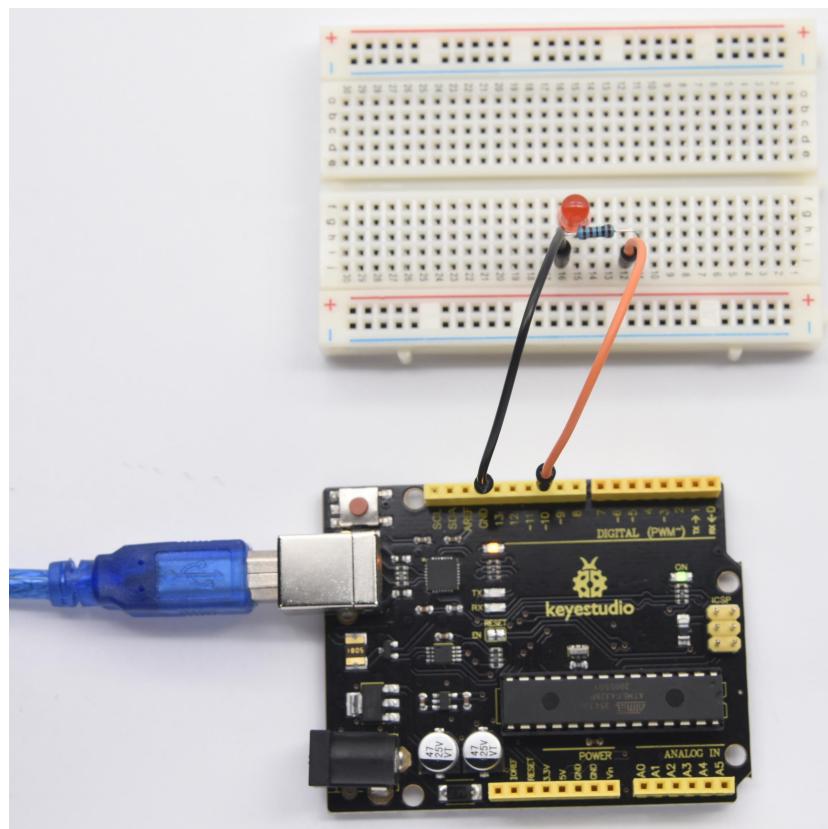
```
}
```

```
||||||||||||||||||||||||||||||||||||||||||
```

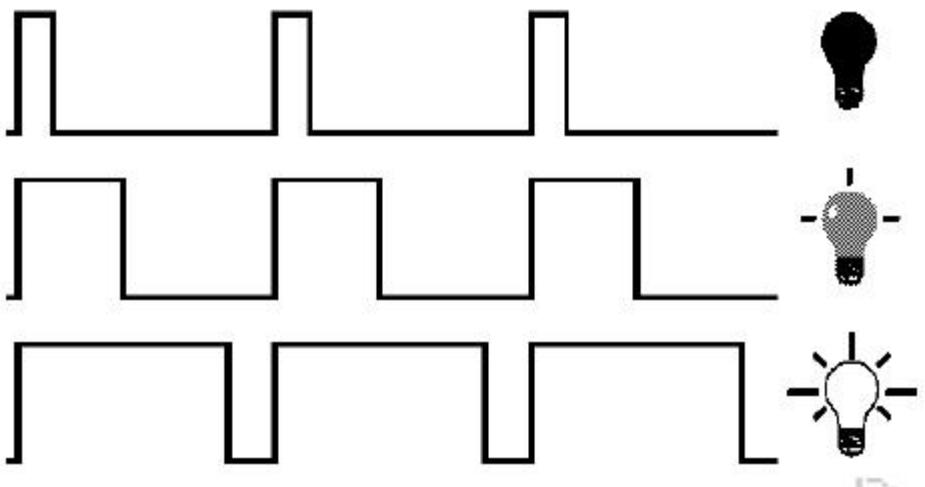
6. результат теста

После загрузки этой программы в эксперименте вы увидите, как светодиод, подключенный к выводу 10, включается и выключается с интервалом примерно в одну секунду.

Таким образом, эксперимент с миганием светодиода завершен. Спасибо!



Проект 3: ШИМ



1. Введение

ШИМ, сокращение от широтно-импульсной модуляции, - это метод, используемый для кодирования уровня аналогового сигнала в цифровой. Компьютер не может выводить аналоговое напряжение, а только цифровые значения напряжения, такие как 0 В или 5 В. Поэтому мы используем счетчик с высоким разрешением для кодирования определенного уровня аналогового сигнала путем модуляции рабочего цикла PWM.

2. принцип работы

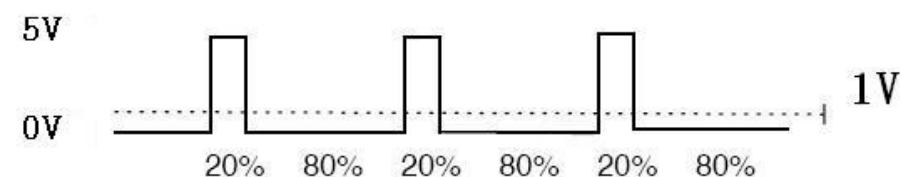
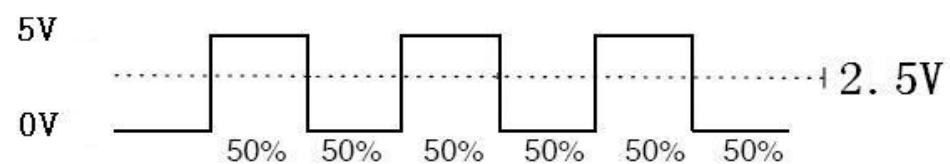
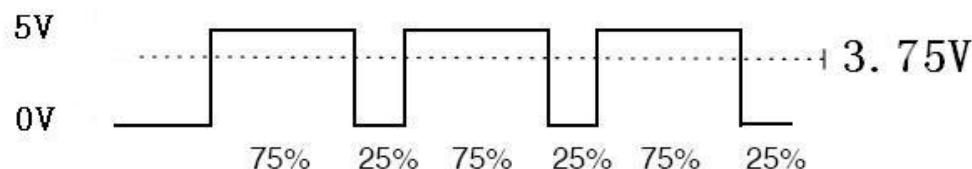
Сигнал PWM также оцифровывается, потому что в любой момент полностью включенный источник питания постоянного тока составляет либо 5 В (ВКЛ), либо 0 В (ВЫКЛ). Напряжение или ток подается на аналоговую нагрузку (устройство

ТОТ использует ТО мощность) по повторяется пульс

последовательность ВКЛ или ВЫКЛ. Во включенном состоянии ток подается на

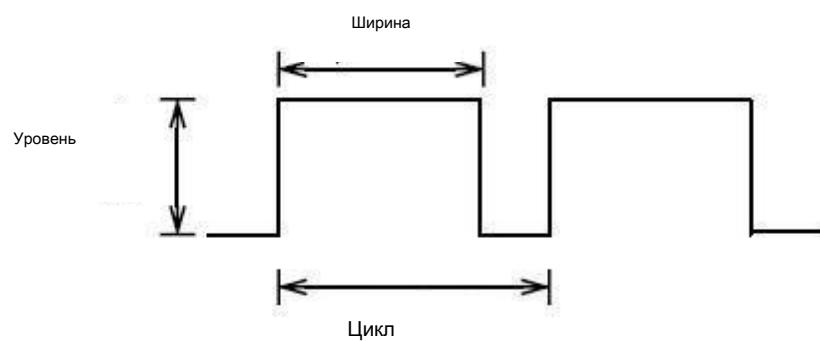
нагрузка; будучи выключенным, это не так. При соответствующей полосе пропускания любое аналоговое значение может быть закодировано с помощью ШИМ. Значение выходного напряжения рассчитывается через время включения и выключения.

Выходное напряжение = (время включения / время импульса) * максимальное значение напряжения



ШИМ имеет множество применений: регулировка яркости лампы, регулировка скорости двигателя, создание звука и т. Д.

Ниже приведены три основных параметра PWM.



1. Амплитуда ширины импульса (минимум / максимум)
2. Период импульса (величина, обратная частоте импульсов в секунду)
3. Уровень напряжения (например, 0–5 В).

На Arduino есть 6 интерфейсов PWM, а именно цифровой контакт 3, 5, 6, 9, 10 и 11.

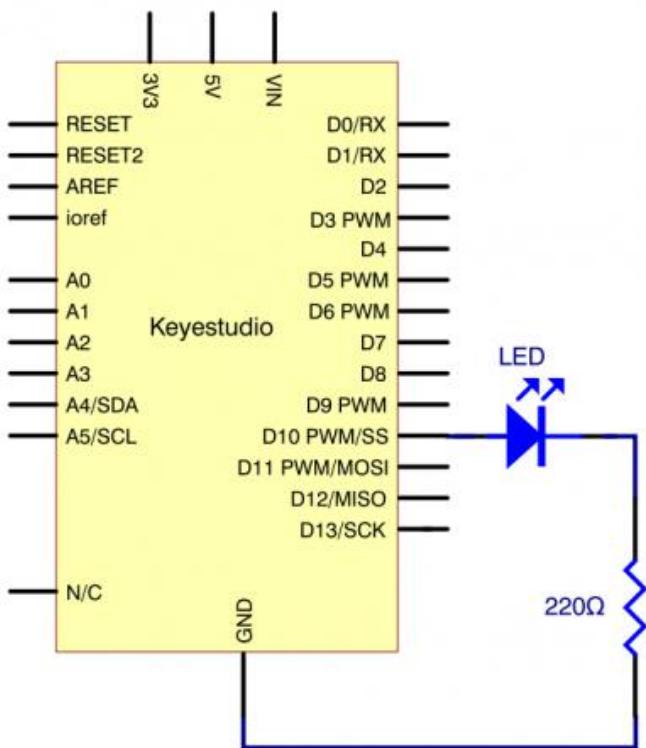
В предыдущих экспериментах мы использовали «светодиод с кнопочным управлением», используя цифровой сигнал для управления цифровым выводом, также как и потенциометр.

На этот раз мы будем использовать потенциометр для управления яркостью светодиода.

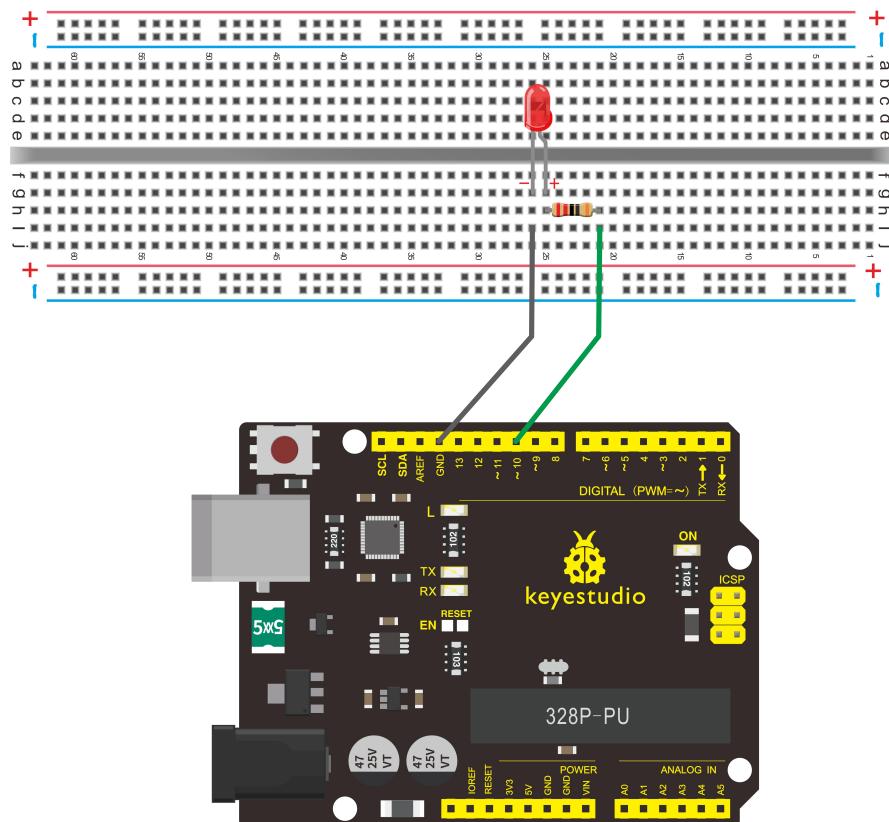
3. требуется оборудование

- Плата V4.0 или плата MEGA 2650 * 1
- Красный светодиод M5 * 1
- 220 Ом резистор
- Макетная плата * 1
- Перемычка макетной платы * 6
- USB-кабель * 1

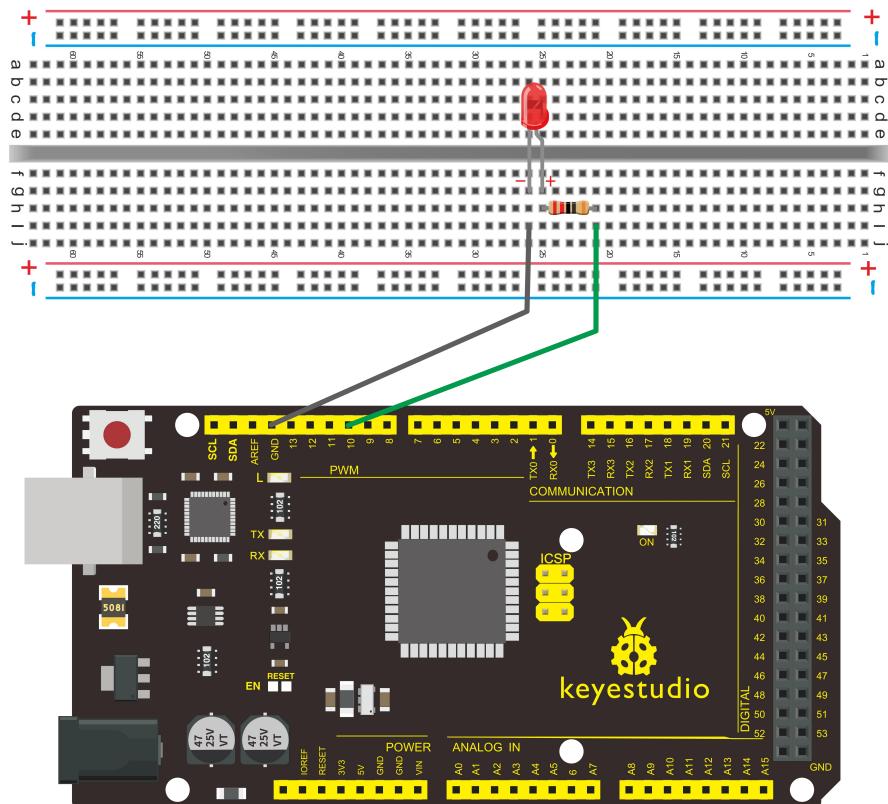
4. схема подключения



Подключение для V4.0



Подключение для 2560 R3



5. образец кода

/*

супер обучающий комплект keyestudio Project

3

ШИМ

<http://www.keyestudio.com>

* /

int ledPin = 10; void

setup () {

pinMode (ledPin, ВЫХОД);

```
}

void loop () {

для (int значение = 0; значение <255; значение = значение + 1) {analogWrite

(ledPin, значение);

задержка (5);

}

для (целое значение = 255; значение> 0; значение = значение-1) {analogWrite

(ledPin, значение);

задержка (5);

}

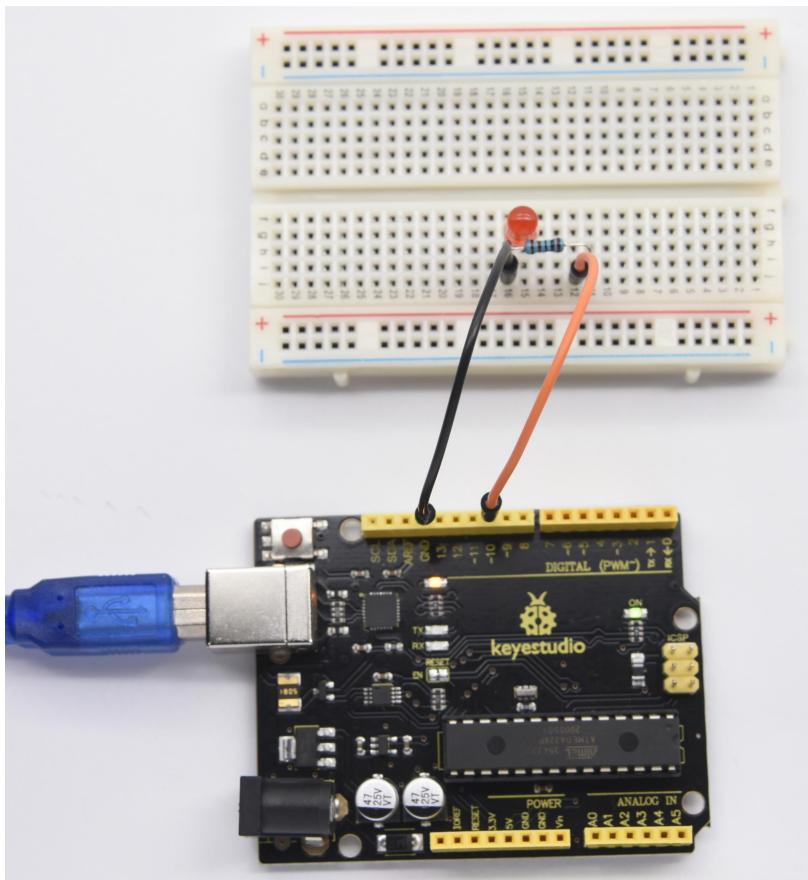
}

||||||||||||||||||||||||||||||||||||
```

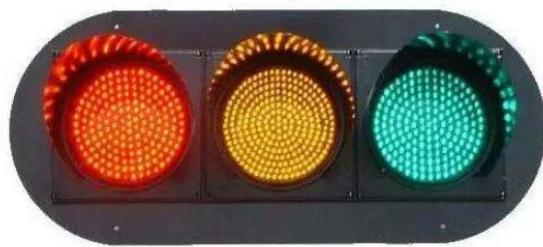
6. результат теста

Подключитесь по схеме подключения, загрузите код и подключите питание.

Внешний светодиод загорается, затем постепенно гаснет, что похоже на дыхание человека.



Проект 4: Светофор



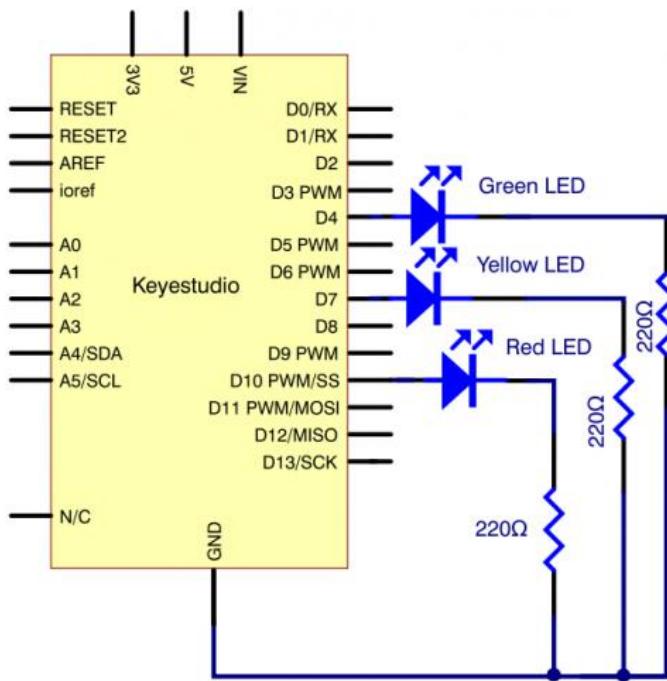
1. Введение

В предыдущей программе мы провели эксперимент с миганием одного светодиода. Пришло время сделать ставку и провести более сложный эксперимент - светофор. Собственно, эти два эксперимента похожи. В этом эксперименте со светофором мы используем три светодиода разного цвета, а не светодиод.

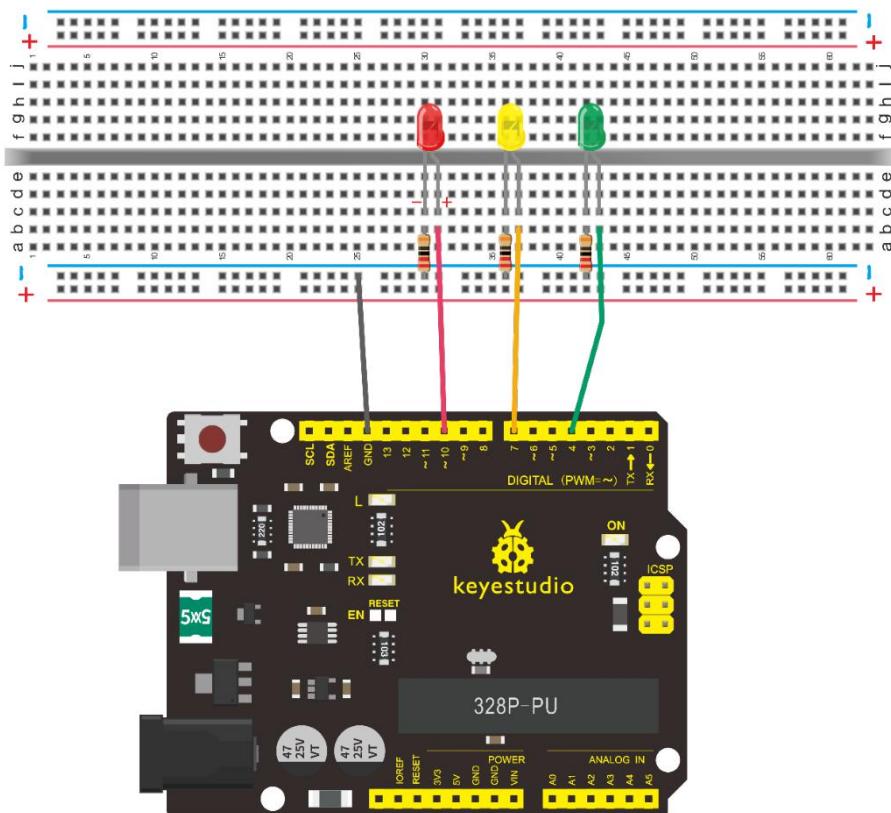
2. Требуется оборудование

- Плата V4.0 или плата MEGA 2650 * 1
- USB-кабель * 1
- Красный светодиод M5 * 1
- Желтый светодиод M5 * 1
- Синий светодиод M5 * 1
- Резистор 220 Ом * 3
- Макетная плата * 1
- Перемычка макетной платы * 4

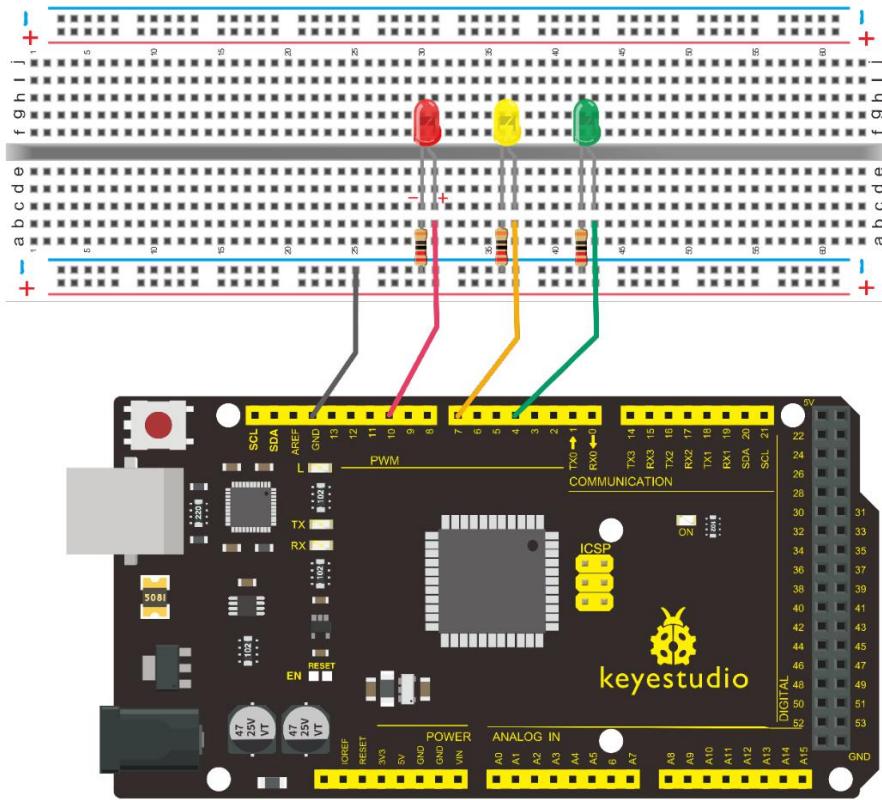
3. схема подключения



Подключение для V4.0



Подключение для 2560



4..Пример кода

Поскольку это имитация светофора, время мигания каждого светодиода

должно быть таким же, как и в системе светофора.

В этой программе мы используем Arduino **задержка ()** функция управления временем

задержки, которая намного проще, чем язык С.

```
/*
```

супер обучающий комплект keyestudio Project

4

[светофор](#)

<http://www.keyestudio.com>

* /

int redled = 10; // инициализируем цифровой контакт 10. int yellowled

= 7; // инициализируем цифровой вывод 7. int blueled = 4; //

инициализируем цифровой вывод 4. void setup ()

{

pinMode (redled, OUTPUT); // устанавливаем вывод с красным светодиодом как «выходной» pinMode

(желтым, OUTPUT); // устанавливаем вывод с желтым светодиодом как «выход»

pinMode (синий, ВЫХОД); // устанавливаем вывод с синим светодиодом как «выход»}

пустой цикл ()

{

digitalWrite (blueled, HIGH); /// включаем синий светодиод delay

(5000); // ждем 5 секунд

digitalWrite (синий, LOW); // выключаем синий светодиод для (int

i = 0; i < 3; i++) // мигает 3 раза {

delay (500); // ждем 0,5 секунды

digitalWrite (yellowled, HIGH); // включаем желтый светодиод delay (500);

// ждем 0,5 секунды

```
digitalWrite (желтые, LOW); // выключить желтый светодиод}

delay (500); // ждем 0,5 секунды

digitalWrite (redled, HIGH); // включаем красный светодиод delay

(5000); // ждем 5 секунд

digitalWrite (redled, LOW); // выключить красный светодиод}
```

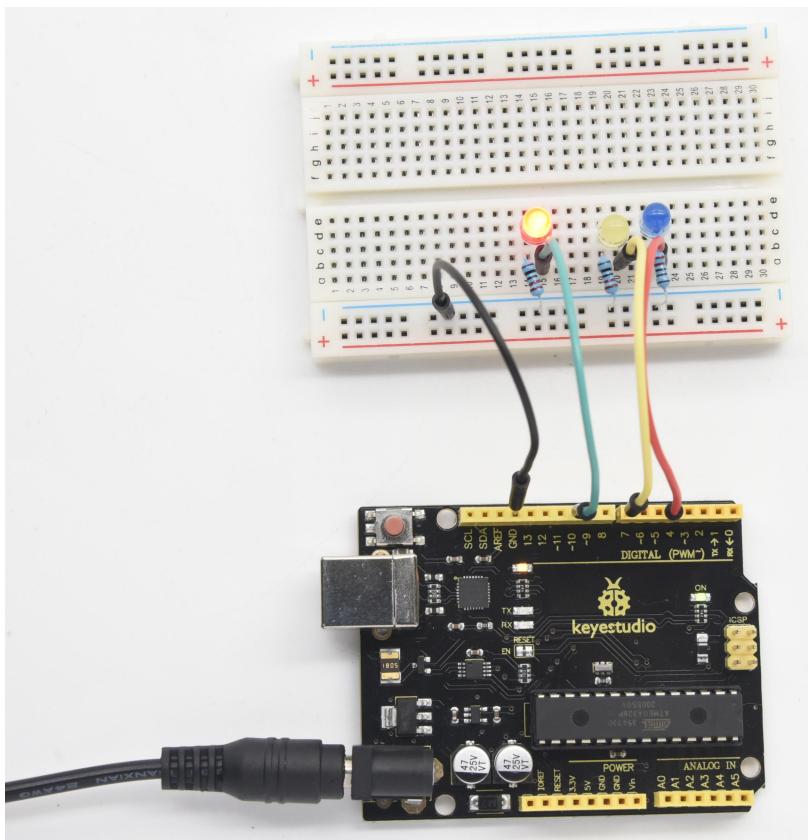
```
||||||||||||||||||||||||||||||||||||||||
```

5. результат теста

Когда процесс загрузки завершится, вы сможете увидеть светофоры собственной конструкции. Обратите внимание, что эта схема очень похожа на схему с эффектом погони за светодиодами.

Синий свет будет гореть 5 секунд, а затем погаснет, затем желтый свет мигнет 3 раза, а затем красный свет загорится на 5 секунд, повторяя цикл.

Эксперимент завершен, спасибо!



Проект 5: Эффект преследования светодиодов



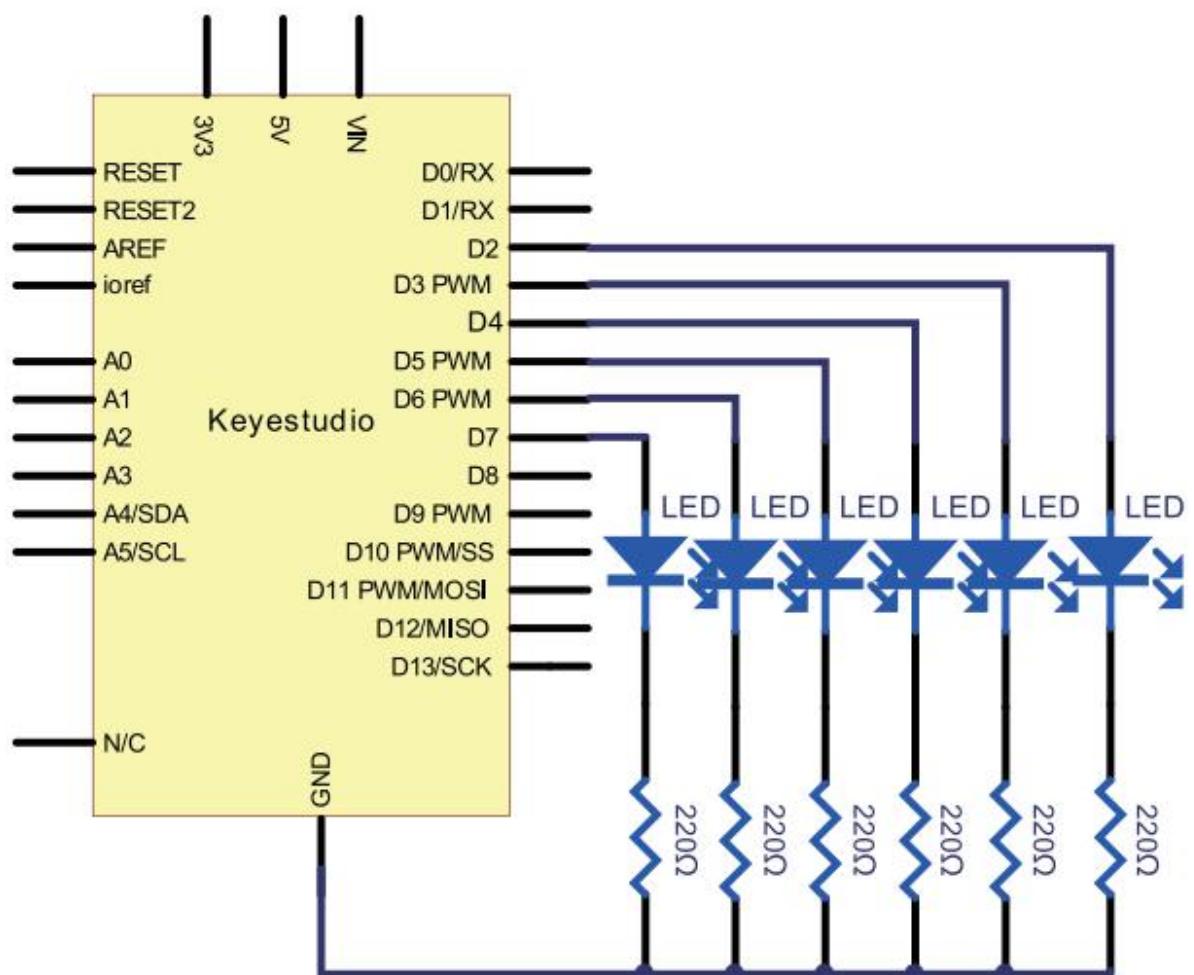
1. Введение

Мы видим множество рекламных щитов, состоящих из разноцветных светодиодов. Они постоянно меняются, чтобы сформировать различные эффекты. В этом эксперименте мы составляем программу для моделирования эффекта погони.

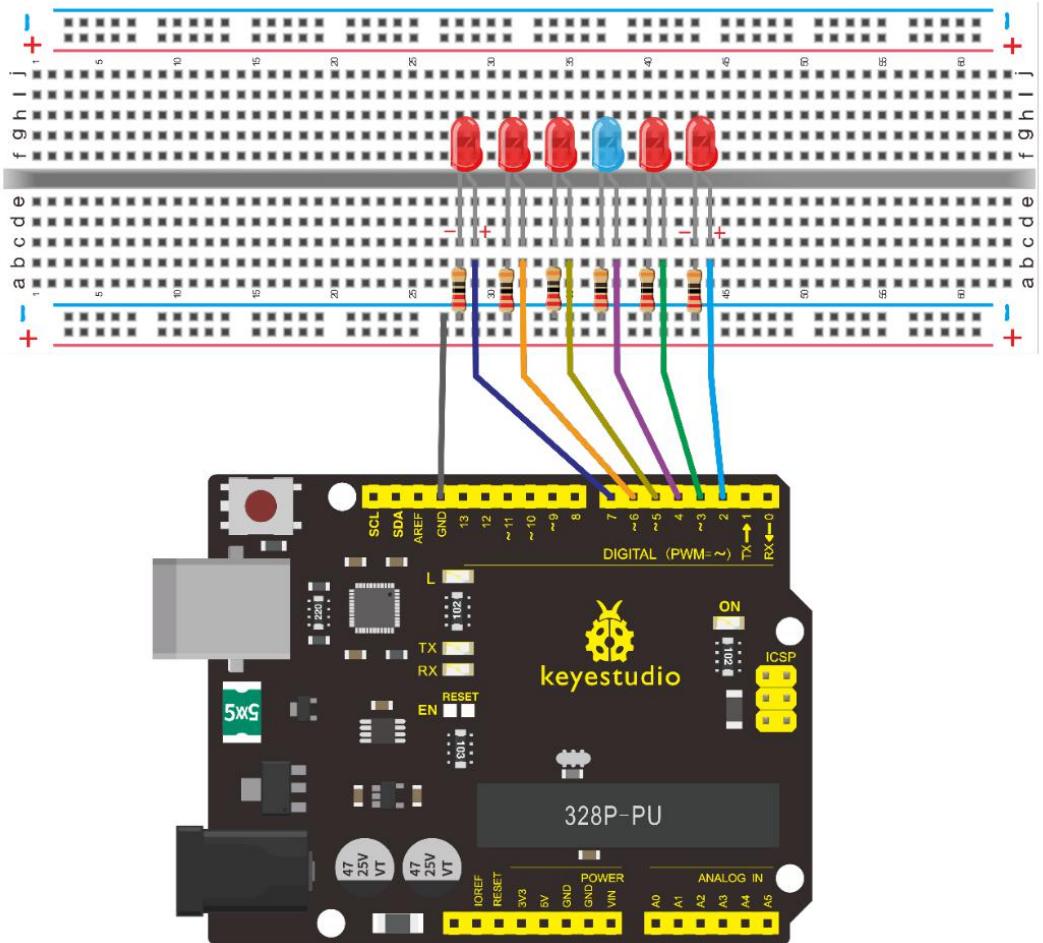
2. Требуется оборудование

- Плата V4.0 или плата MEGA 2650 * 1
- Красный светодиод * 6
- Резистор 220 Ом * 6
- Перемычка макетной платы * 12
- USB-кабель * 1

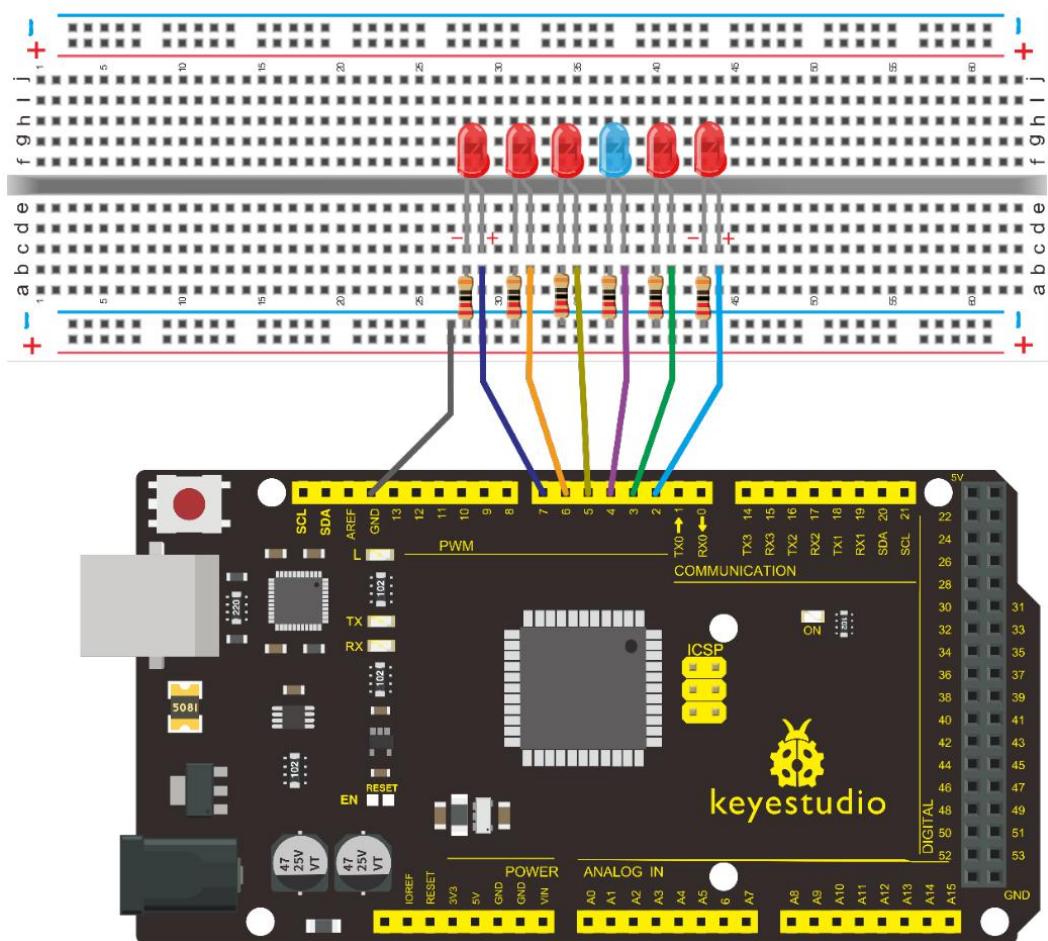
3. схема подключения



Подключение для V4.0



Подключение для 2560



4. образец кода

/*

супер обучающий комплект keyestudio Project

5

Светодиодный эффект погони

<http://www.keyestudio.com>

* /

```
int BASE = 2; // вывод ввода-вывода для первого светодиода int NUM =
```

```
6; // количество светодиодов
```

```

установка void () {  

    for (int i = BASE; i <BASE + NUM; i++) {  

        pinMode (я, ВЫХОД);           // устанавливаем выводы ввода / вывода как вывод  

    }  

}  

пустой цикл () {  

    for (int i = BASE; i <BASE + NUM; i++) {  

        digitalWrite (я, LOW);         // установить контакты ввода / вывода на «низкий», выключить светодиоды один  

        ОДНИМ.  

        задержка (200);             // задержка  

    }  

    for (int i = BASE; i <BASE + NUM; i++) {  

        digitalWrite (я, ВЫСОКИЙ);    // устанавливаем контакты ввода / вывода на высокий уровень, включаем светодиоды  

        по одному  

        задержка (200);             // задержка  

    }  

}

```