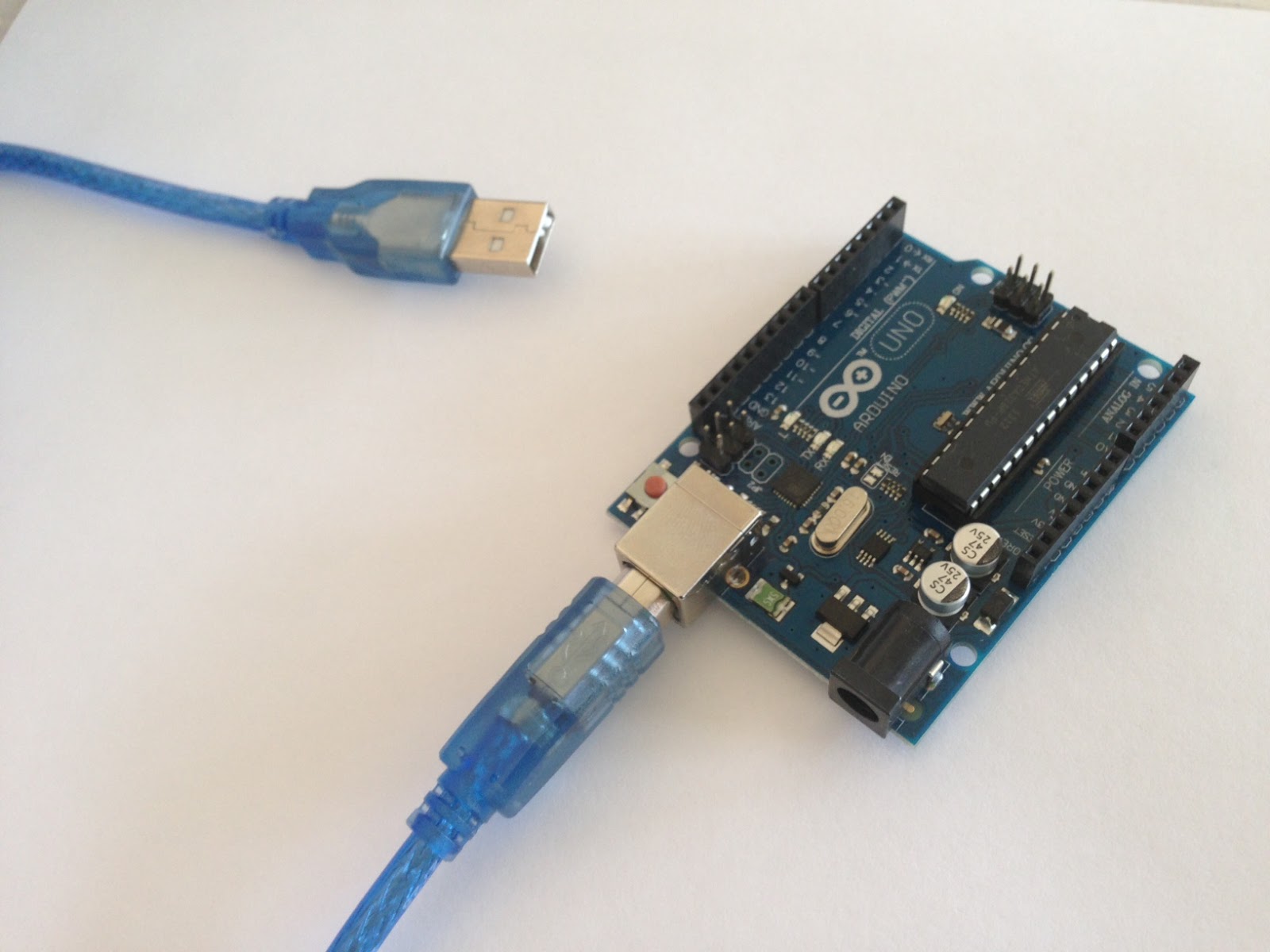
## Знакомство с микроконтроллером Arduino Uno

Краткие сведения: Arduino (ардуино) — популярная аппаратная вычислительная платформа, основными компонентами которой являются плата ввода-вывода и среда разработки. Arduino может использоваться как для создания автономных интерактивных объектов, роботов, так и подключаться к программному обеспечению, выполняемому на компьютере. Платы имеют аналоговые и цифровые порты, к которым можно подключить большое количество датчиков и исполнительных устройств: кнопка, помпа, мотор, светодиод, экран и т.д.

Изучение Arduino часто входит в учебные программы по робототехнике и изучается в кружках робототехники. Но плата популярна и в создании «настоящих» роботов. Именно Arduino может стать «мозгом» вашего робота.  
Для начала, мы проверим работоспособность Arduino на примере мигания светодиода.

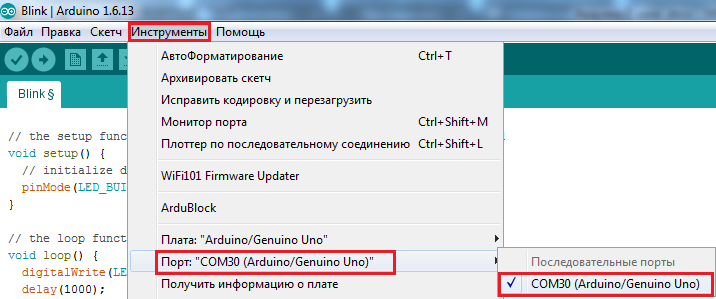
## Подключение микроконтроллера Arduino Uno к компьютеру

Для того чтобы подключить Arduino Uno к компьютеру необходимо использовать кабель USB 2.0 Type-B.

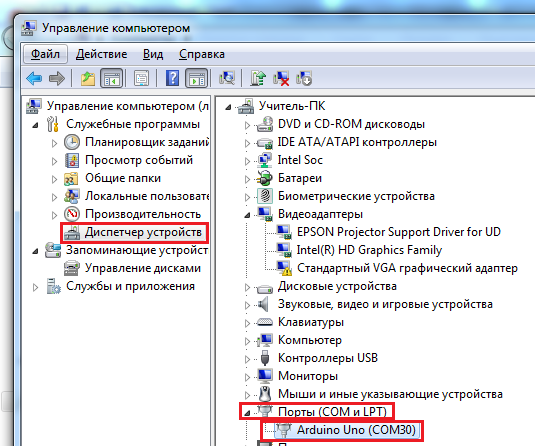


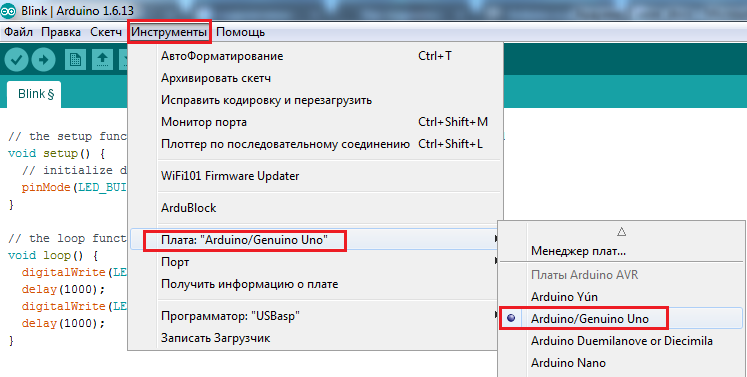
После того, как вы подключили Arduino к компьютеру и открыли Arduino IDE, вам необходимо установить в программе нужный COM порт, и тип используемой платы.

Для этого в главном меню вы нажимаете на вкладку Инструменты, выбираете порт и устанавливаете тот COM, к которому подключен ваш микроконтроллер.

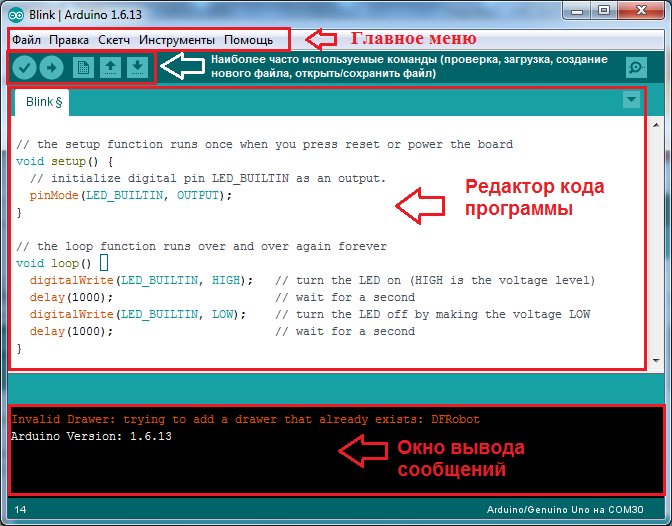


Если пункт меню «Порт» не активен, проверьте, правильно ли подключен кабель. Если вы не знаете, к какому COM порту подключена ваша плата, посмотрите в диспетчере устройств.



Далее необходимо выбрать нужную плату: Инструменты/Плата/Arduino/Genuino Uno

## Знакомство со средой программирования Arduino IDE

****

Главное меню обеспечивает доступ ко всем функциям среды Arduino IDE.

Под строкой главного меню находятся кнопки наиболее часто используемых команд.

Редактор кода автоматически выделяет цветом различные части программы (переменные, процедуры, функции и т.д.).

Окно вывода сообщений: вывод информационных сообщений во время компиляции, загрузки программы и т.д.

## Создание программы мигания встроенным светодиодом

Контроллер Arduino UNO уже содержит резистор и LED-светодиод, подключенный к 13 выводу, поэтому никаких других внешних радиоэлементов нам не понадобится.

Для того чтобы ваш светодиод начал мигать, вам необходимо открыть соответствующую программу из примеров, которые уже есть в самом языке программирования Arduino.

Для этого в верхнем меню вы выбираете Файл/Примеры/Basics/Blink, либо же вы можете скопировать программу ниже:

/\*

\* Lesson1\_1

\* Мигание LED

\* ------------

\*

\* Включает и выключает светодиод (LED) подсоединенный

\* к выходу 13, с интервалом в 2 секунды

\*

\*/

int ledPin = 13; // LED подсоединен к выводу 13

void setup()

{

pinMode(ledPin, OUTPUT); // устанавливаем вывод 13 как выход

}

void loop()

{

digitalWrite(ledPin, HIGH); // включаем LED

delay(1000); // пауза 1 секунда (1000 мс)

digitalWrite(ledPin, LOW); // выключаем LED

delay(1000); // пауза 1 секунда (1000 мс)

}

|  |  |
| --- | --- |
|  |  |

После подачи напряжения на Arduino автоматически запускается процедура *setup*, которая устанавливает вывод 13 как выход. Затем автоматически в цикле запускается процедура *loop*, которая включает и выключает светодиод с паузой в 1 секунду.

После того, как программа откомпилируется и загрузится в микроконтроллер – ваш светодиод должен мигать. Если этого не происходит, убедитесь, что вы нигде не допустили ошибок.

*Для загрузки программы в микроконтроллер Arduino нажмите кнопку «Загрузка» на панели инструментов *

Если программа не загружается, попробуйте отключить Arduino и подключить заново.

**Задание:** Попробуйте изменить промежуток времени между включением и выключением светодиода.

**Решение:** Функция *delay* отвечает за паузу (промежуток между включением и выключением светодиода), попробуйте изменить значения в этой функции.

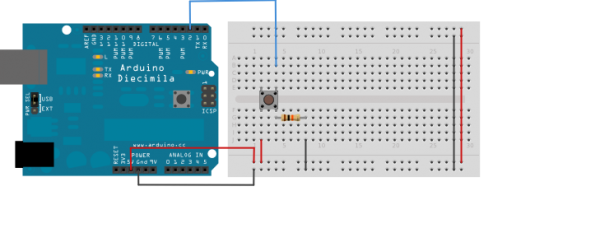
## Подключение кнопки к Arduino Uno

Теперь, когда все разобрались со светодиодом, можно немного усложнить задачу. Теперь светодиод будет включаться и выключаться не сам по себе, а при помощи «кнопки».

**Необходимые компоненты**

* контроллер Arduino
* тактовая кнопка
* 10кОм резистор
* breadboard
* соединительные провода

**Подключение**

[](http://arduino.ru/sites/default/files/tutorials/Example_button.png)

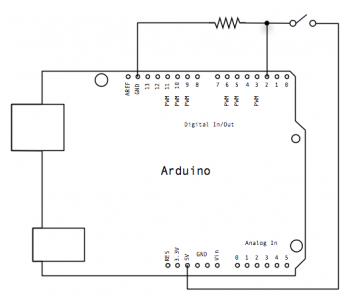
*Примечание: мигает светодиод с подписью L.*

Подключаем выход питания (5V) и землю (Gnd), красным и черным проводом к макетной плате. Обычно на макетных платах для питания и земли используют крайние ряды контактов, как показано на рисунке. Третьим синим проводом мы соединяем цифровой пин 2 контроллера Arduino к контакту тактовой кнопки. К этому же контакту, либо к контакту, постоянно соединенному с ней в 4х штырковом исполнении, подключаем подтягивающий резистор 10 кОм, который в свою очередь соединяем с землей. Другой выход кнопки соединяем с питанием 5 В.

Когда тактовая кнопка не нажата, выход 2 подключен только к земле через подтягивающий резистор и на этом входе будет считываться [LOW](http://arduino.ru/Reference/Constants). А когда кнопка нажата, появляется контакт между входом 2 и питанием 5В, и считываться будет [HIGH](http://arduino.ru/Reference/Constants).

*Замечание: Чаще всего тактовые кнопки имеют по два контакта с каждой стороны так, как это показано на рисунке подключение. При этом по форме кнопка почти квадратная. ВАЖНО не перепутать при подключении какие контакты соединены, а какие замыкаются при нажатие. Лучше всего прозвонить кнопку, если вы не уверены.*

Можно также подключить кнопку наоборот — через подтягивающий резистор к питанию и через кнопку к земле. Тогда с входа будет считываться HIGH, а при нажатие кнопки, LOW.

**Схема**[](http://arduino.ru/sites/default/files/tutorials/Example_button_scheme.png)

Если вход оставить неподключенным, то на входе будет считываться HIGH или LOW случайным образом. Именно поэтому мы используем подтягивающий резистор, чтобы задать определенное значение при ненажатой кнопке.

## Создание программы управления работой светодиода при помощи кнопки

**Программный код lesson1\_2**

/\*

\* lesson 1\_2

Управление работой светодиода с помощью кнопки

\*/

// задаем константы

const int buttonPin = 2; // номер входа, подключенный к кнопке

const int ledPin = 13; // номер выхода светодиода

// переменные

int buttonState = 0; // переменная для хранения состояния кнопки

void setup() {

// инициализируем пин, подключенный к светодиоду, как выход

pinMode(ledPin, OUTPUT);

// инициализируем пин, подключенный к кнопке, как вход

pinMode(buttonPin, INPUT);

}

void loop(){

// считываем значения с входа кнопки

buttonState = digitalRead(buttonPin);

// проверяем нажата ли кнопка

// если нажата, то buttonState будет HIGH:

if (buttonState == HIGH) {

// включаем светодиод

digitalWrite(ledPin, HIGH);

}

else {

// выключаем светодиод

digitalWrite(ledPin, LOW);

}

}

После загрузки программы, когда вы нажимаете кнопку, то светодиод включен, а когда кнопка в ненажатом состоянии, то светодиод выключен.

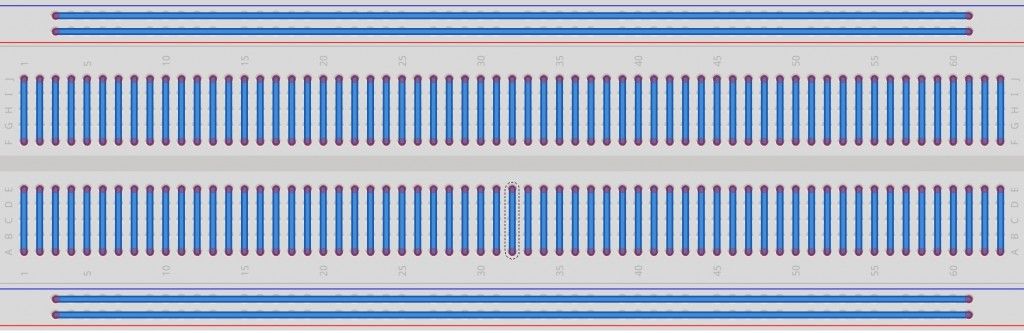
## Подключение внешнего светодиода к Arduino Uno

Если у вас есть необходимые компоненты, то вы можете подключить резистор через макетную плату (breadboard).

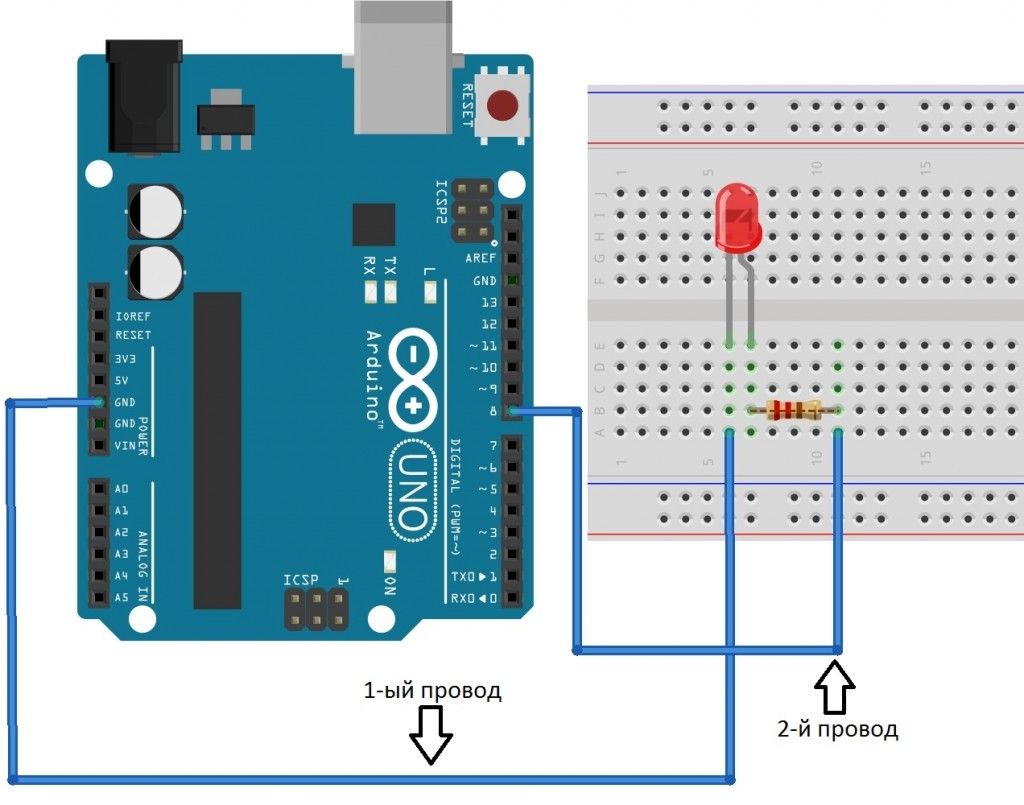
Для подключения мигающего светодиода на Arduino и управления им вам понадобится:

* плата Arduino;
* breadboard;
* 2 провода «папа-папа»;
* светодиод;
* резистор.

Breadboard представляет из себя сетку из гнезд, которые обычно соединяются так:



для удобства приводим схему подключения светодиода на Arduino:



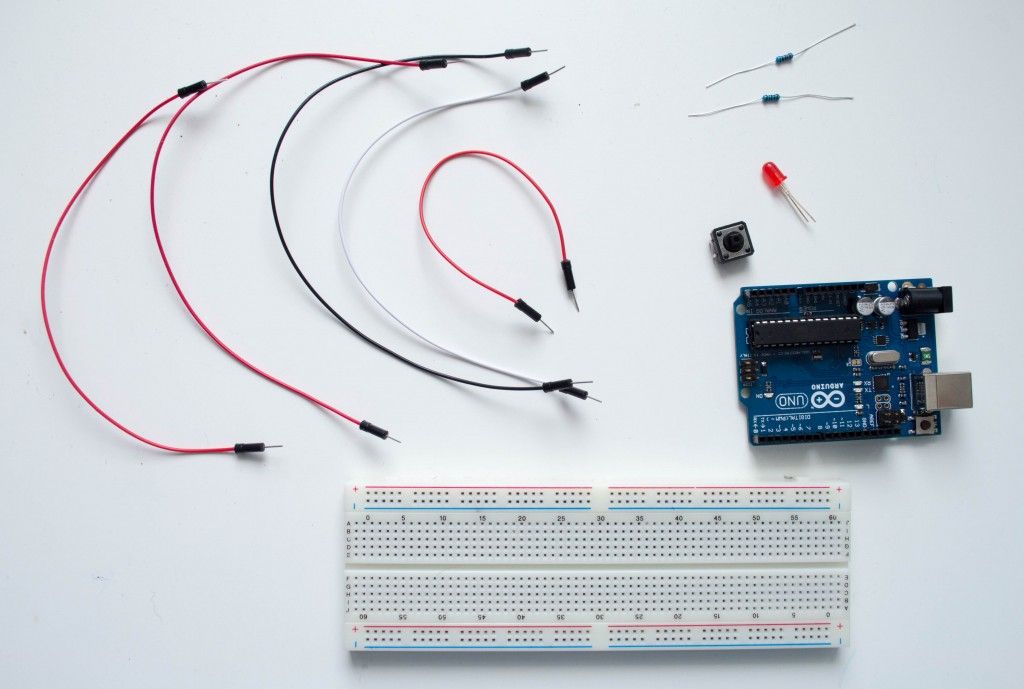
*Примечание: Длинная ножка светодиода – плюс, короткая – минус.*

Для работы этой модели подойдет программа Lesson1\_1.

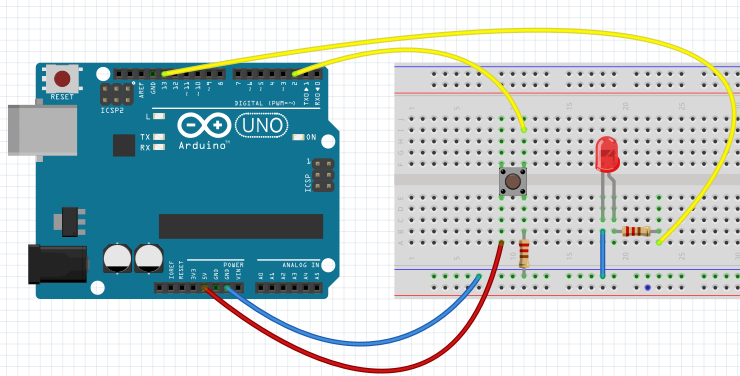
После того, как программа откомпилируется и загрузится в микроконтроллер – ваш светодиод должен мигать. Если этого не происходит, убедитесь, что вы правильно собрали схему и не допустили ошибок в коде.

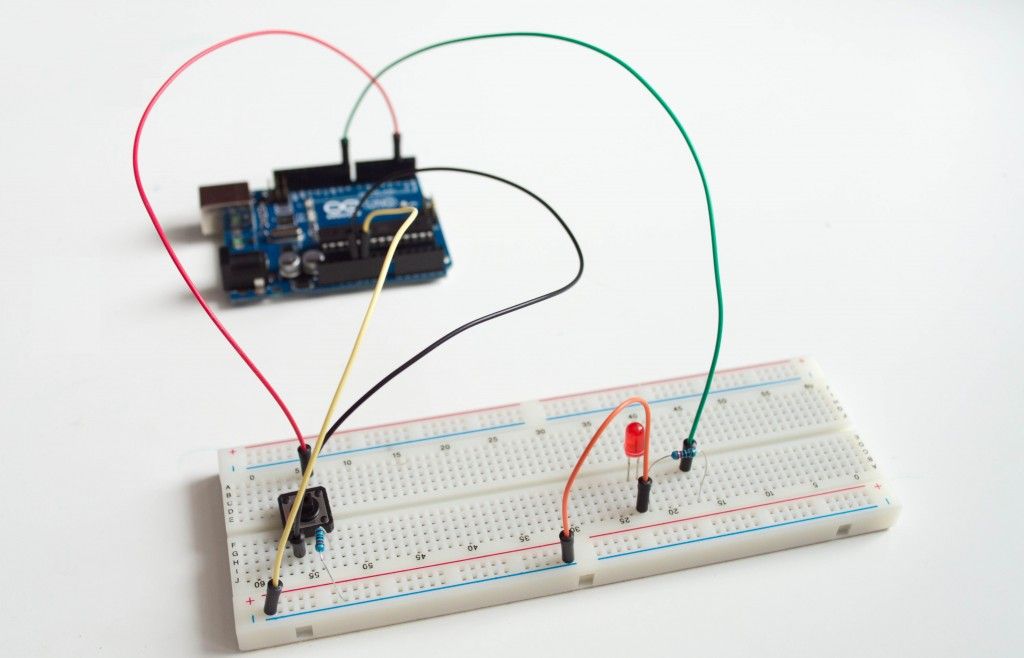
## Управление работой внешнего светодиода при помощи кнопки

Можно подключить внешний светодиод при помощи кнопки так же, как мы это делали со светодиодом, встроенным в микроконтроллер.

[](http://edurobots.ru/wp-content/uploads/2014/03/%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%BD%D0%B5%D0%BD%D1%82%D1%8B1.jpg)Для сборки модели нам потребуется:

* плата Arduino;
* Breadboard;
* 5 проводов и/или перемычек «папа-папа»;
* светодиод;
* кнопка;
* резисторы на 10 кОм и 220 Ом.

Схема подключения модели Arduino с кнопкой и светодиодом:

Так выглядит собранная модель Arduino кнопки со светодиодом:

Для работы этой модели подойдет программа Lesson1\_2.