

# Arduino и бионика

Введение в микроконтроллеры с Arduino

## Занятие 1



11 ноября 2007 - machineproject — Тод Е. Курт

Перевод на русский язык

16 сентября 2011 — robofreak.ru — Татьяна Волкова

# Информация о занятиях

- Раздаётся флэшка, на ней:
  - PDF-версия этой презентации
  - Среда Arduino для MacOSX и Windows
  - Исходные тексты («скетчи») примеров
  - Скопируйте и передайте дальше
- Воскресенье: 3 часа
  - две части по ~1.5 ч. с перерывом 15 м.
- Четверг: ~2.5 часа
  - с небольшим повторением в начале

# Программа на сегодня

- Введение в Arduino
- Установка среды разработки
- Ваш первый скетч на Arduino
- Цифровые и аналоговые выходы
- Цифровые входы
- Как помигать светодиодом
- Как воспринять нажатие кнопки  
и узнать состояние переключателя

# Бионика?

Электронные органы чувств заменяют человеческие?

Искусственные "мышцы" работают как естественные?  
Или лучше?

Что, недоступное человеку, могут почувствовать  
электронные устройства?

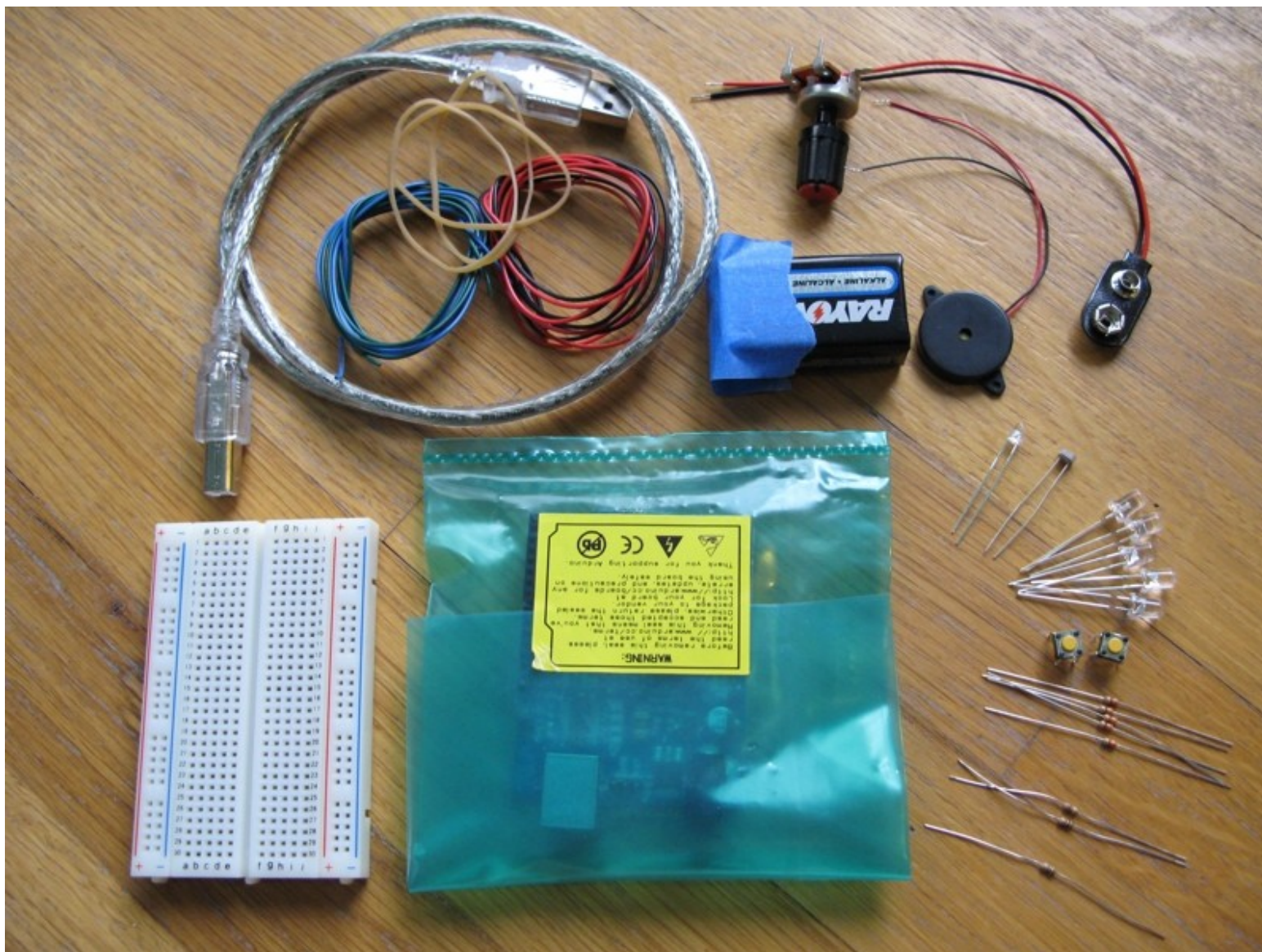
Как бы вы улучшили себя с новыми возможностями?

В этих занятиях при помощи простой платы Arduino  
изучаются входные и выходные компоненты,  
используемые в роботах, мобильных телефонах,  
компьютерных играх и автомобилях.

Ваши устройства наблюдают за вами и отвечают вам -  
знайте их слабые места на случай восстания машин.

В худшем случае, вы хотя бы сможете выдать себя за своего, обклеив тело платами Arduino.

# Содержимое набора для занятия 1



Набор для занятия 2 будет на следующей неделе

Небольшая коробочка (например, из-под обуви) отлично подойдёт для хранения деталей.  
Не показаны RGB-светодиоды. Упс. Они появились уже после фотосъёмки.

# Набор для занятия 1

Минимум необходимого, и «звук и свет»

- Плата Arduino Diecimila USB
- беспаяечная макетная плата
- USB-кабель
- пьезоизлучатель
- потенциометр
- 5 оранжевых светодиодов (большие, прозрачные)
- 1 RGB-светодиод (матовый, общий анод)
- две тактовые кнопки
- батарейка 9В с клеммами
- резисторы:
  - 6 x 220 ом (красный-красный-коричневый)
  - 2 x 10 кОм (коричневый-чёрный-оранжевый)
  - 1 x 1 МОм (коричневый-чёрный-зелёный)
- фоторезистор
- фототранзистор (маленький, прозрачный)
- провода четырёх цветов
- резинки

На следующей неделе - дополнение: "движение и моторы"

# Немного о безопасности

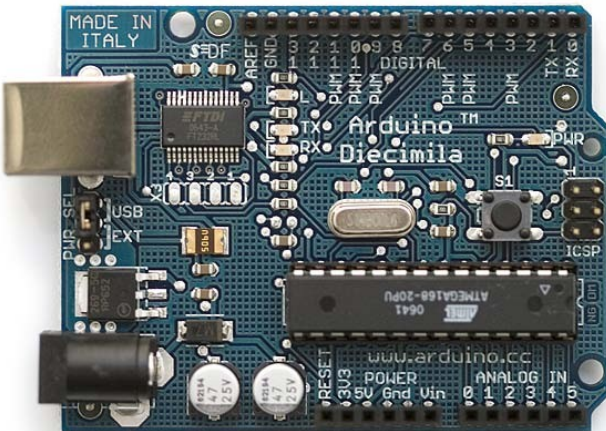
- Электроника может навредить вам
  - Некоторые детали содержат свинец
  - После - обязательное мытьё рук
- Вы можете навредить электронике
  - Чувствительна к статическому электричеству: не шаркайте ногами
  - Только провода можно сгибать без особых последствий



# Что такое Arduino?

У слова "Arduino" 3 значения:

Физическое устройство



Среда разработки

```
Arduino - 0010 Alpha

Blink §

* The basic Arduino example. Turns on an LED on for one second,
* then off for one second, and so on... We use pin 13 because,
* depending on your Arduino board, it has either a built-in LED
* or a built-in resistor so that you need only an LED.
*
* http://www.arduino.cc/en/Tutorial/Blink
*/

int ledPin = 13;           // LED connected to digital pin 13

void setup()               // run once, when the sketch starts
{
  pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop()                // run over and over again
{
  digitalWrite(ledPin, HIGH); // sets the LED on
  delay(1000);                // waits for a second
  digitalWrite(ledPin, LOW);  // sets the LED off
  delay(1000);                // waits for a second
}
```

Сообщество и философия





# Философия и сообщество Arduino

- Аппаратная вычислительная платформа Open Source

- "железо" с открытыми исходниками

- открыто для изучения и изменения

- Аппаратная вычислительная платформа...  
э, что это?

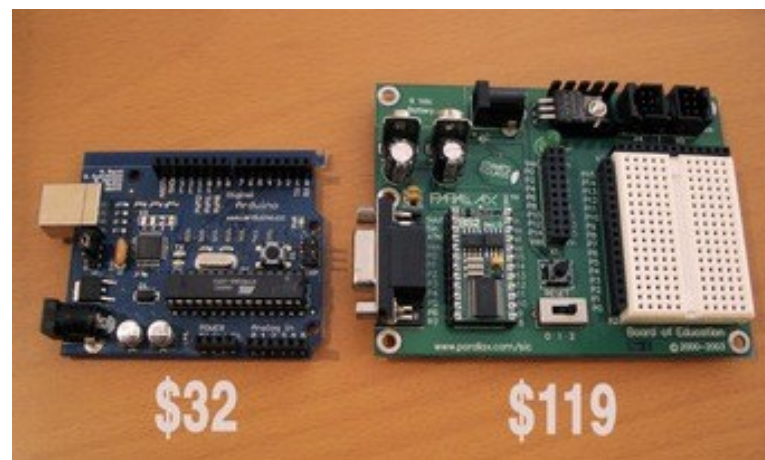
"дополненная реальность" (ubiquitous computing)  
"повсеместная компьютеризация" (pervasive computing)  
"окружающий разум" (ambient intelligence)  
"фоновые вычисления" (calm computing)  
"единое информационное поле" (everyware)  
"живые вещи" (spimes)  
"говорящие вещи" (blogjects)  
"умные вещи" (smart objects)

- Делается сообществом

- Вики с примерами («playground»), которую может править любой
- Форумы, где всегда готовы помочь

# Arduino - аппаратная часть

- Похоже на Basic Stamp (если слышали о таком)
  - но быстрее, дешевле и с открытыми исходниками
- Использует микроконтроллер AVR ATmega168
  - микросхема, разработанная для программирования на языке C



Разработчики AVR специально оптимизировали регистры и набор инструкций так, чтобы программы на C компилировались для неё эффективно. Это большое преимущество в сравнении с прошлыми моделями микроконтроллеров, в которых программы на C были почти всегда менее эффективны, чем вручную закодированные на варианте ассемблера

# Разнообразие плат Arduino

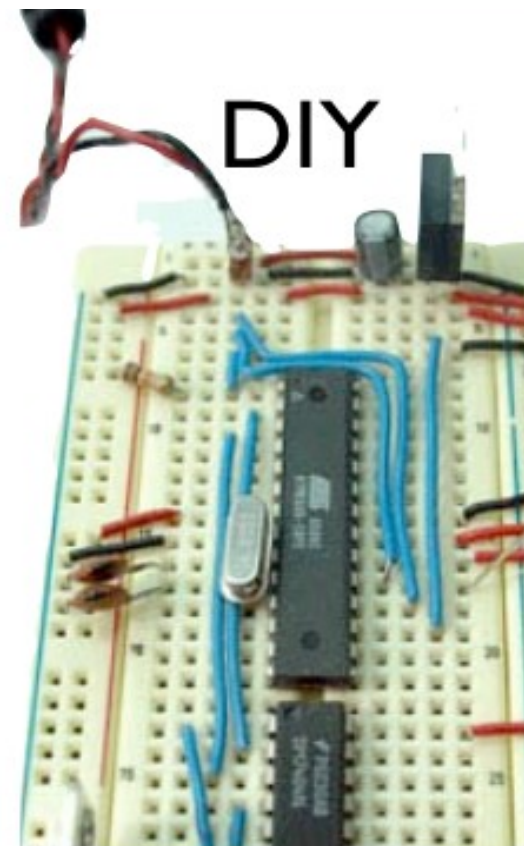
LilyPad  
(для одежды)



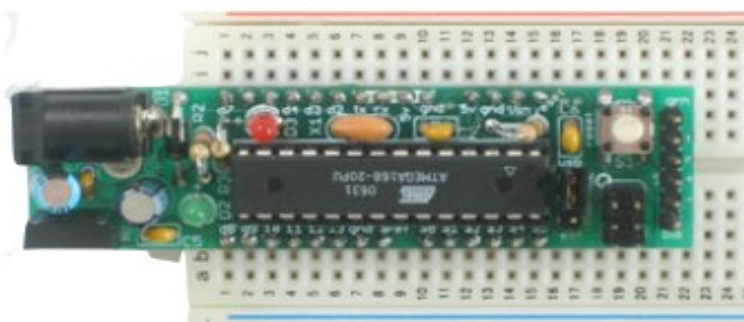
USB



DIY



набор Boarduino



Bluetooth



в «Stamp»-  
формате



множество вариантов для разных нужд

Одно из преимуществ открытой архитектуры - разнообразие вариантов  
Любой может собрать свою плату Arduino, в любом форм-факторе  
Изображения плат взяты со Sparkfun.com и Adafruit.com

# Возможности Arduino

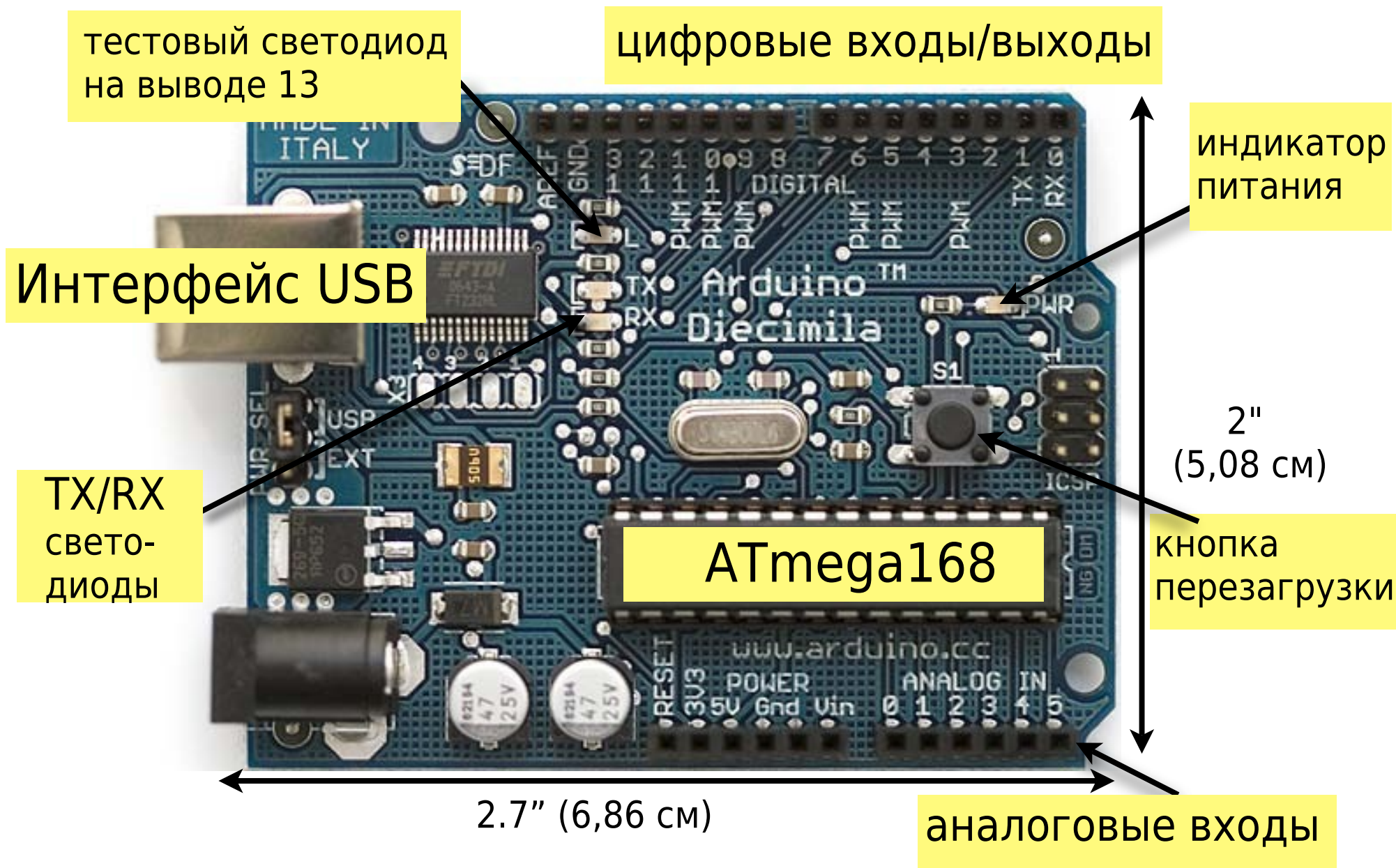
- 16 кБ флэш-памяти программ
- 1 кБ оперативной памяти
- 16 МГц (Apple II: 1 МГц)
- Входы и выходы
  - 13 цифровых входов/выходов
  - 5 аналоговых входов
  - 6 аналоговых выходов\*
- Полностью автономна: однажды запрограммированная, не нуждается в компьютере

\* не совсем аналоговых: используется ШИМ, о которой мы поговорим позже

Не волнуйтесь, если вышесказанное для Вас непонятно - Вы не обязаны это знать.



# Плата Arduino Diecimila



# Терминология Arduino

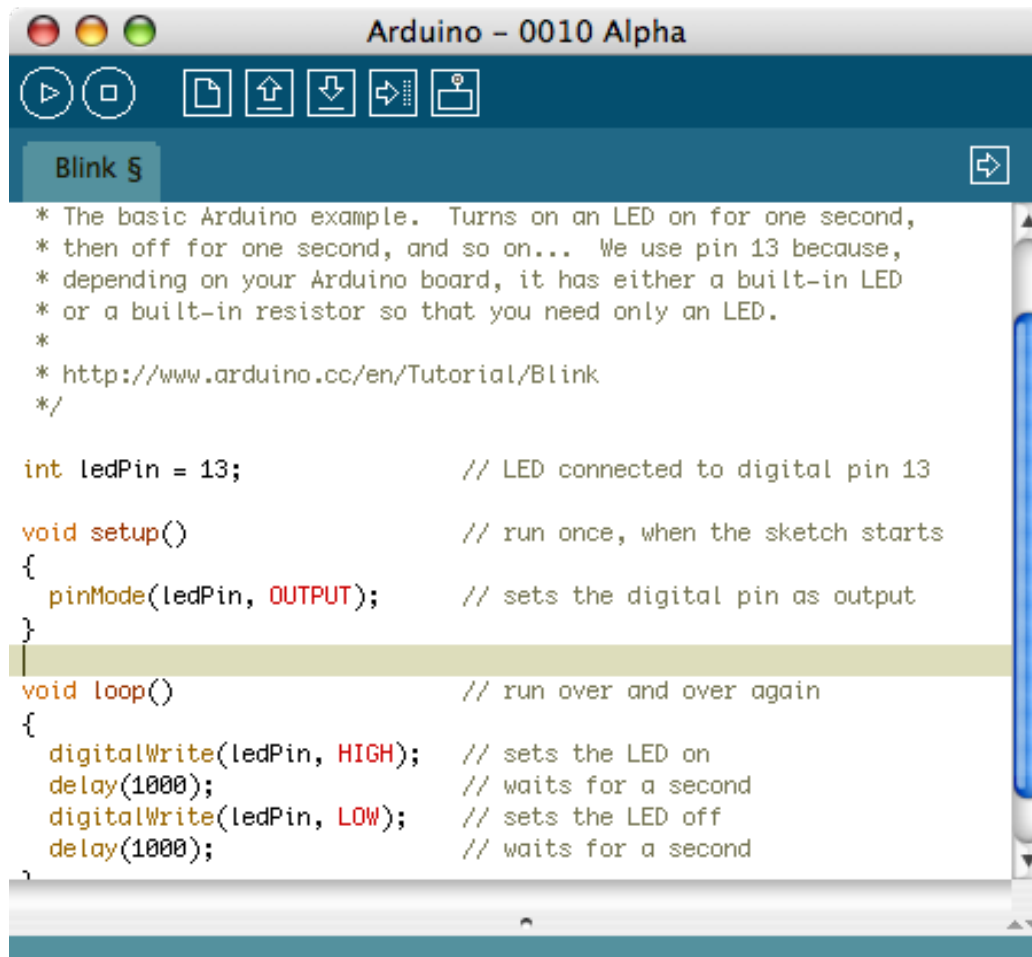
«скетч» - программа, написанная для запуска на Arduino

«ПИН» - вход или выход, подключённый к чему-либо. Пример: светодиод на выходе или кнопка на входе

«цифровой» - значение HIGH или LOW (как вкл/выкл или один/ноль)  
Пример: состояние выключателя

«аналоговый» - значение обычно в промежутке 0-255. Пример: яркость светодиода, скорость мотора.

# Среда программирования Arduino



- Как текстовый редактор
- Можно смотреть/писать/редактировать скетчи
- Но затем Вы загружаете их в микроконтроллер

Если Вы использовали Processing для написания маленьких программ на Java, то заметите, что интерфейс очень похож. Arduino использует графический интерфейс пользователя из Processing и частично разделяет его философию, но код для Arduino и код на Processing - совершенно разные вещи.



# Установка Arduino

## Шаги

1. Раздобыть и разархивировать программу Arduino
2. Подключить плату Arduino
3. Установить драйвер
4. Перезагрузить компьютер
5. Запустить среду Arduino
6. Сообщить Arduino (программе)  
про Arduino (плату)

# Получение и распаковка

- На флэшках
  - «arduino-0010-win.zip» для Windows
  - «arduino-0010-mac.zip» для Mac OS X
- Разархивируйте zip-файл.



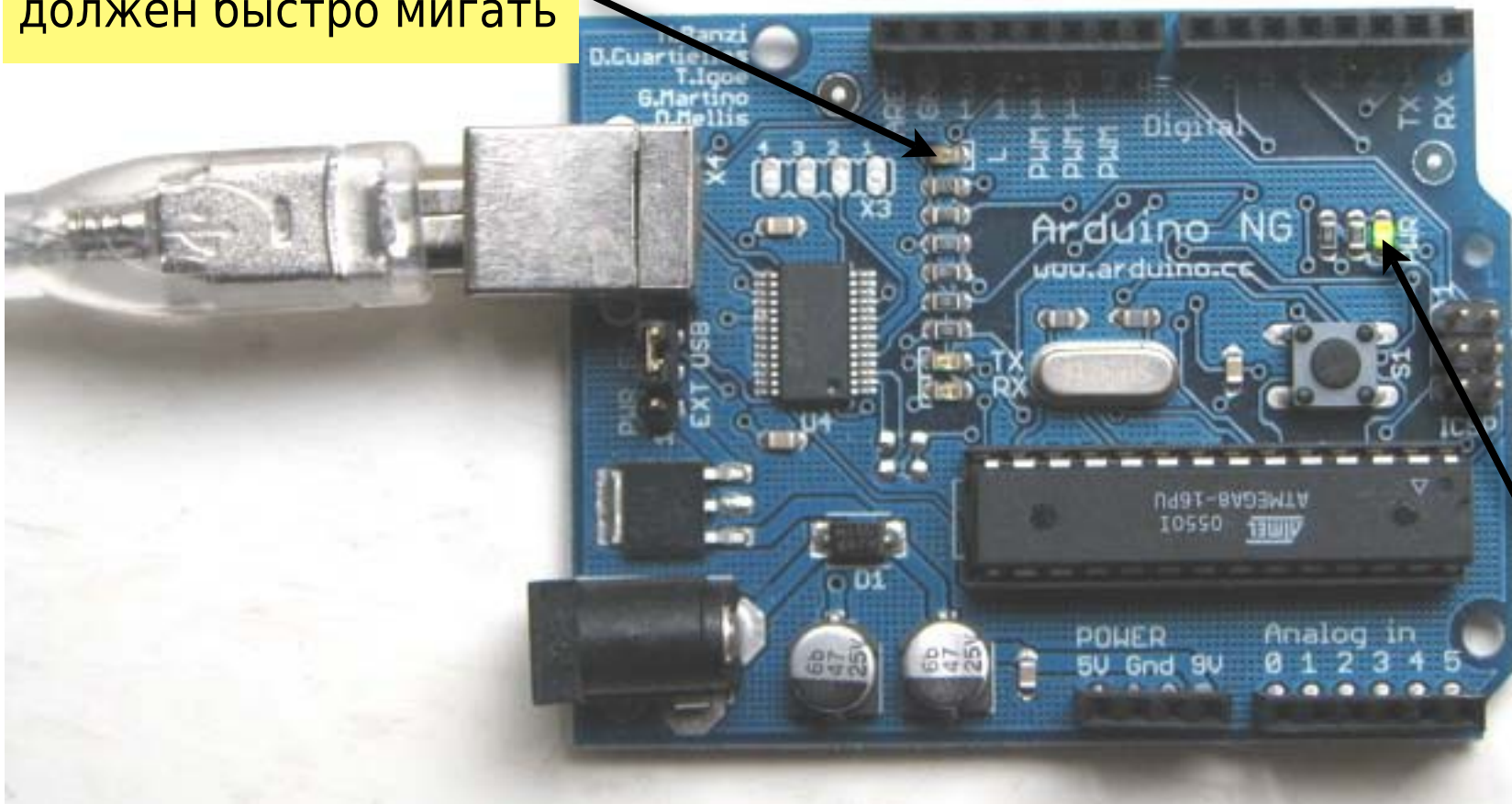
Двойной клик в Mac.  
В Windows правый клик.

«Извлечь всё»

- Внутри найдите папку «drivers»

# Подключение платы Arduino

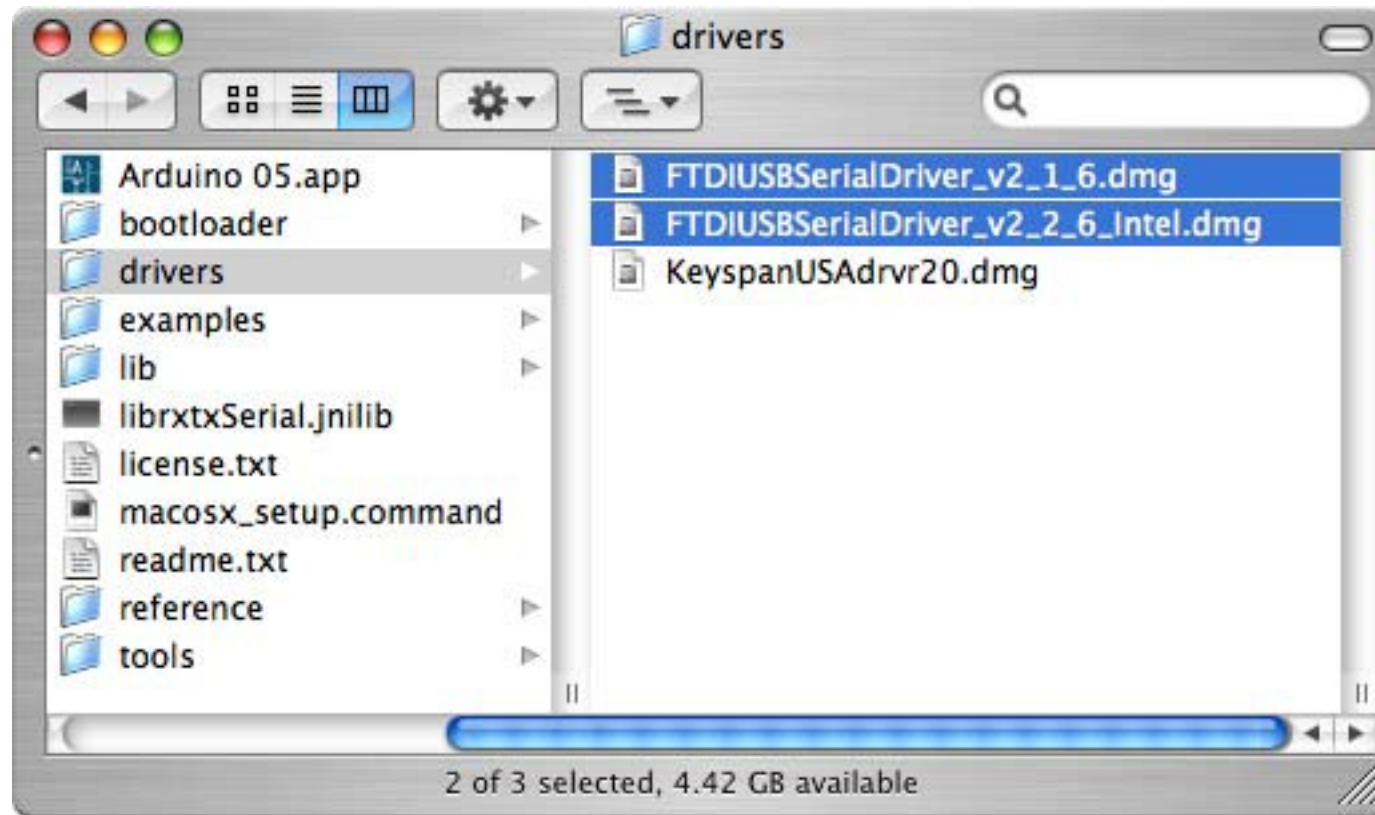
тестовый светодиод  
должен быстро мигать



Индикатор питания должен светиться

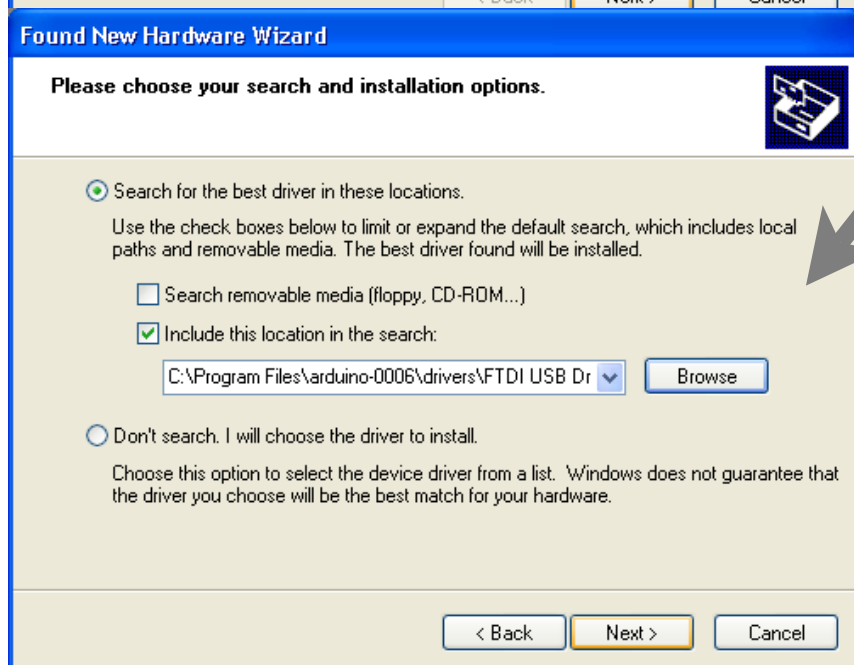
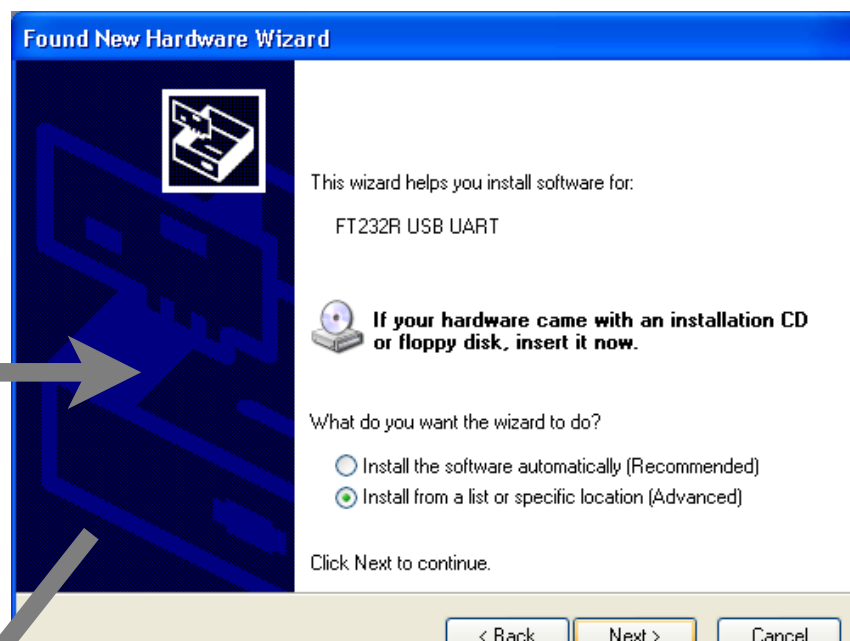
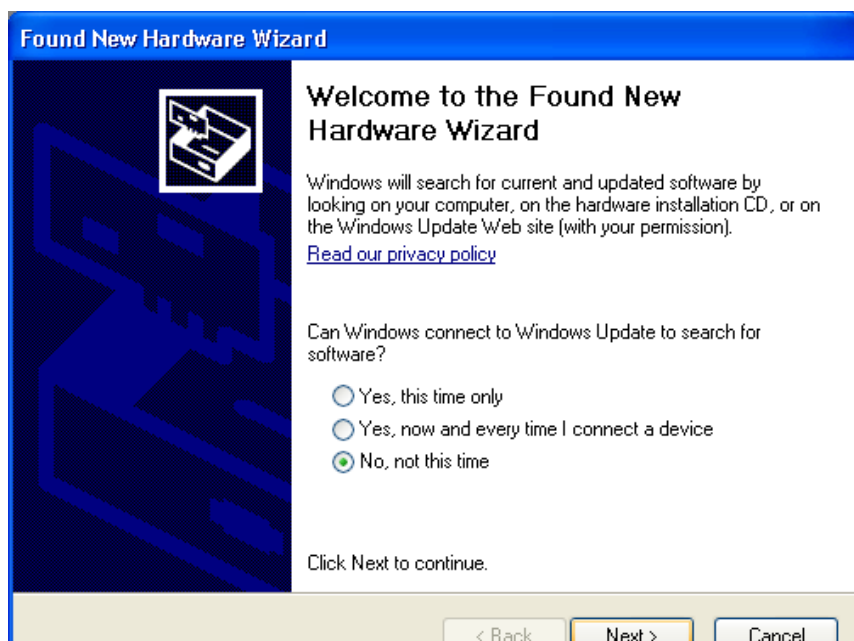
# Установка драйвера под Mac

Двойной клик по установщику .dmg

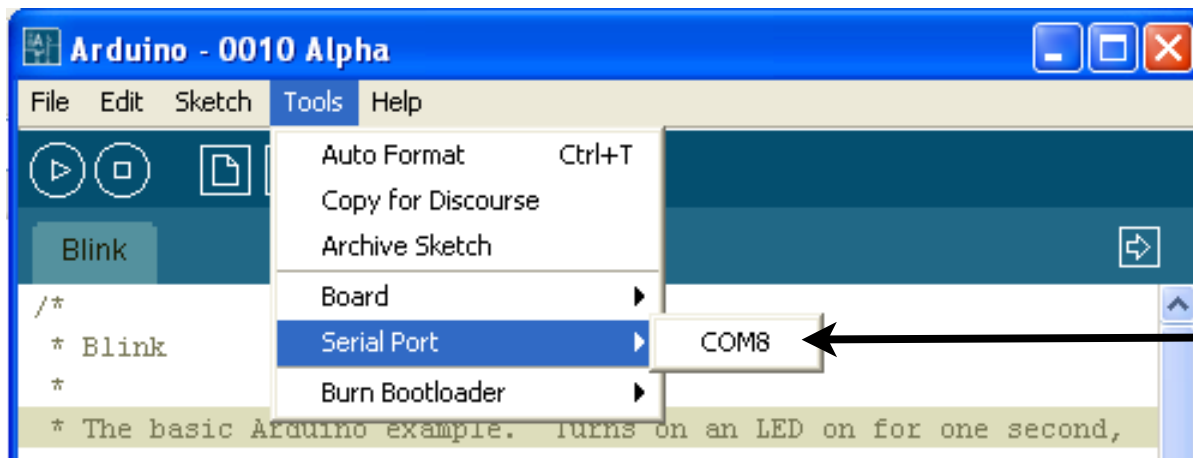


- v2\_1\_6 для PPC Mac
- v2\_2\_6 для Intel Mac

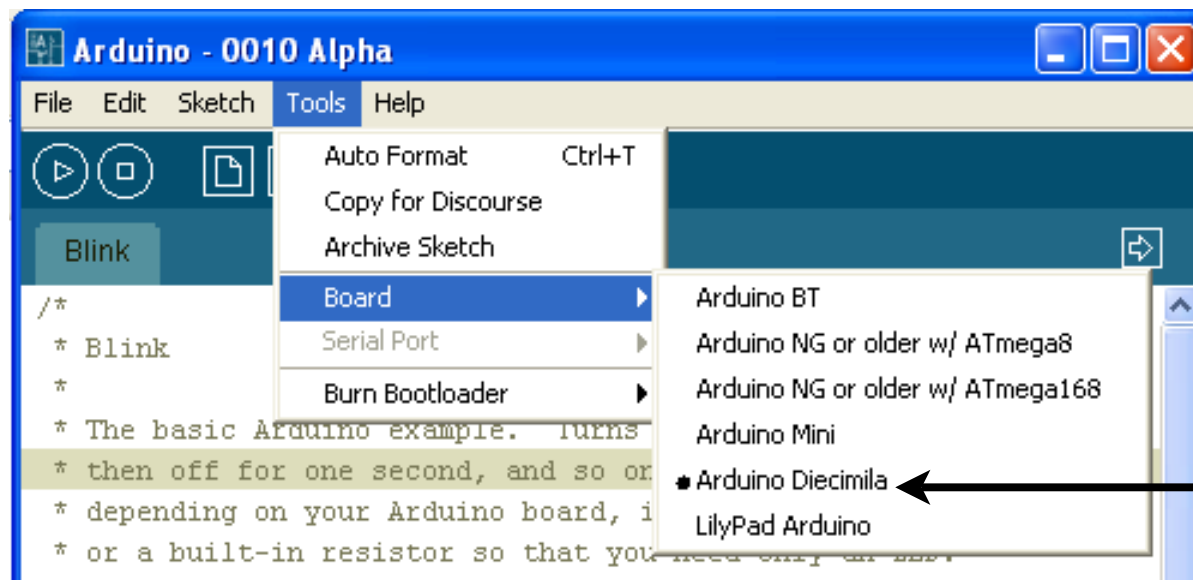
# Установка драйвера под Windows



# Выбор расположения и типа



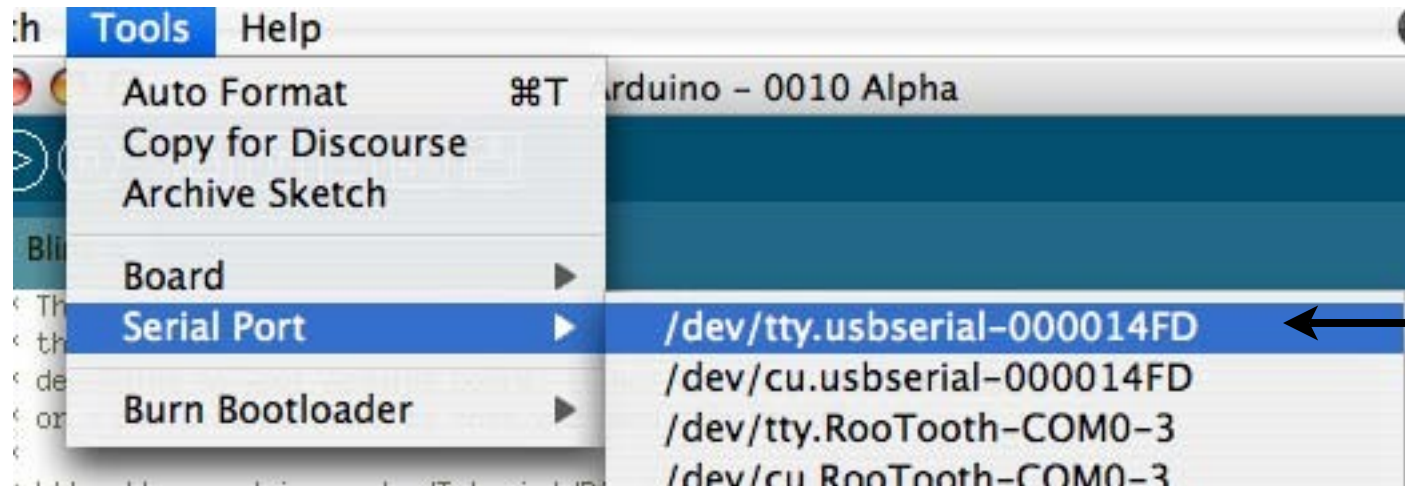
обычно порт с самым  
большим номером



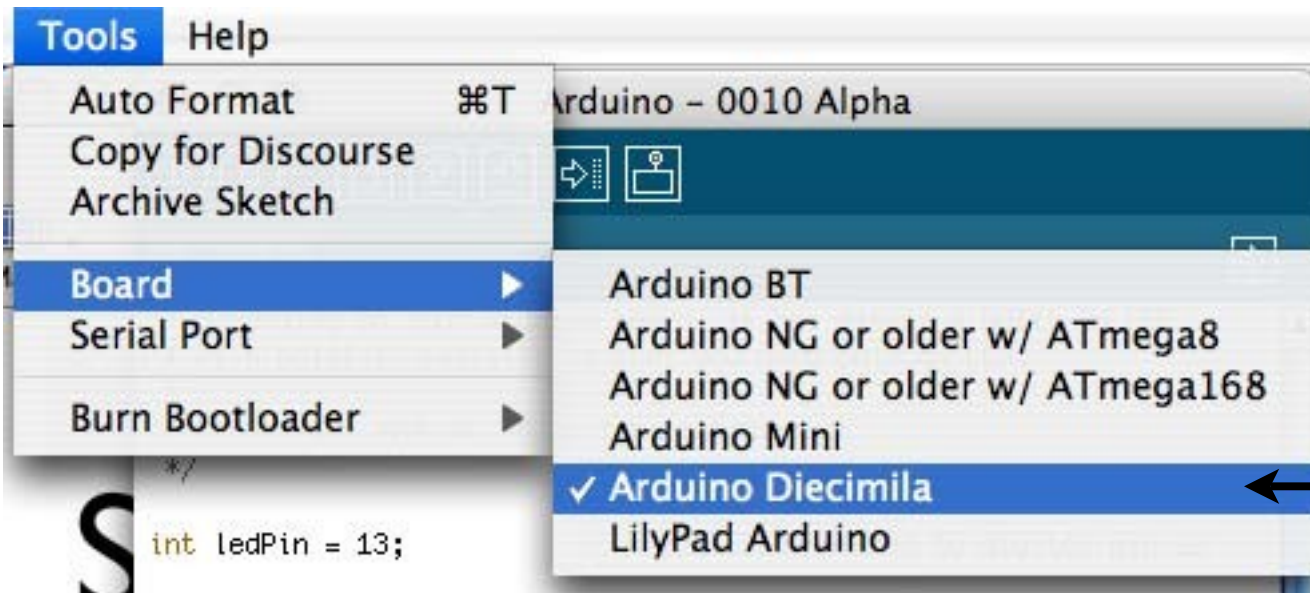
выберите «Diecimila»



# Выбор расположения и типа



начинается  
с *tty.usbserial-*



выберите «Diecimila»

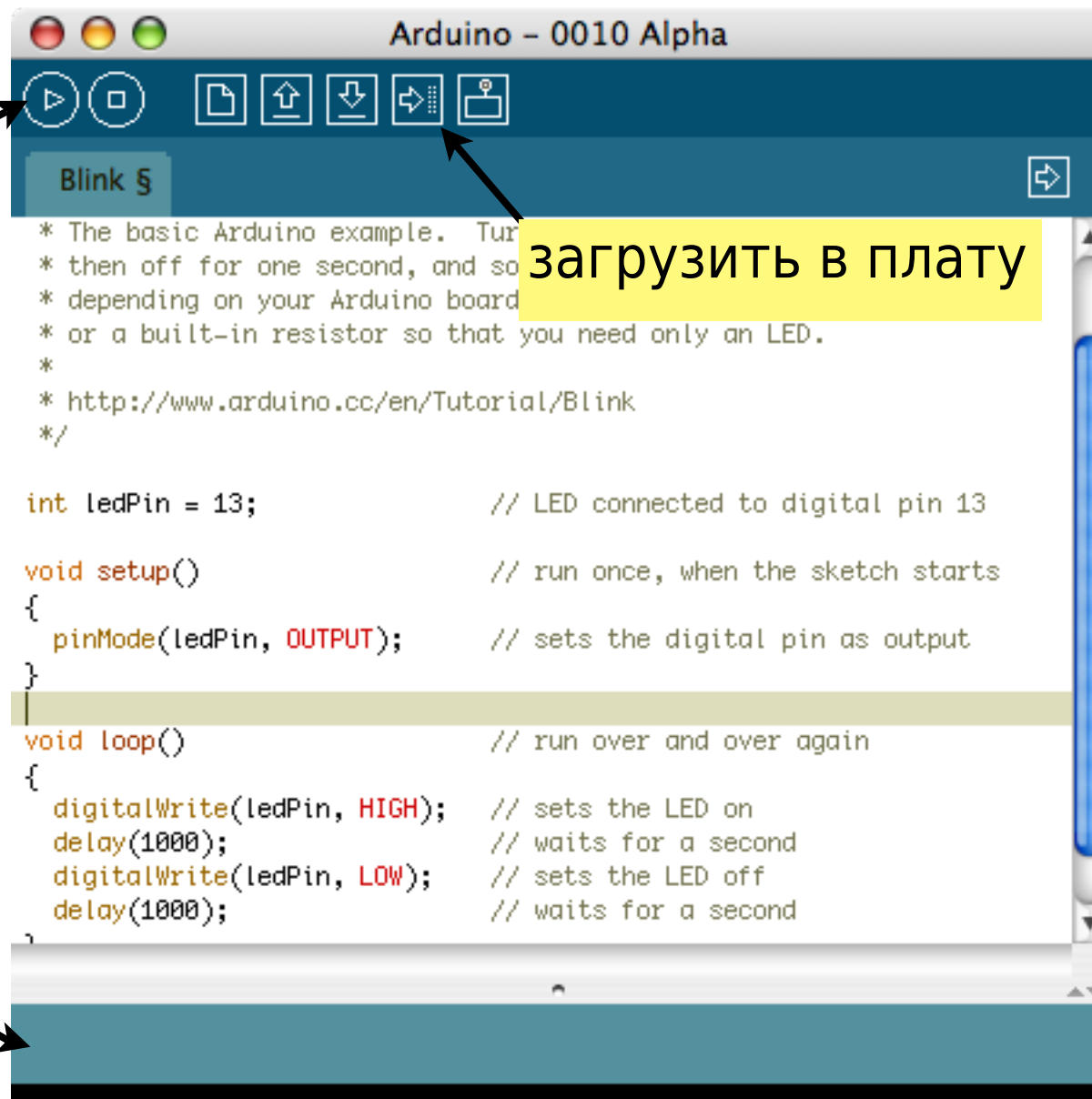


# Среда Arduino

компиляция  
(проверка)

загрузить в плату

статусные  
сообщения



# Использование Arduino

- Напишите свой скетч
- Нажмите кнопку Compile (чтобы проверить на ошибки)
- Нажмите кнопку Upload, чтобы загрузить скетч в плату

Попробуйте скетч «Blink»!

Загрузите  
«File/Sketchbook/Examples/Digital/Blink»

```
void setup() {  
  pinMode(ledPin, OUTPUT); // sets t  
}  
void loop() {  
  digitalWrite(ledPin, HIGH); // sets t  
  delay(1000); // waits  
  digitalWrite(ledPin, LOW); // sets t  
  delay(1000); // waits  
}
```



КОМПИЛЯЦИЯ

Done compiling.



загрузка



TX/RX мигают



скетч  
стартует

Измените значение в "delay()" чтобы изменить частоту мигания

# Статусные сообщения

Загрузка успешна

Done uploading.

Binary sketch size: 1110 bytes (of a 14336 byte maximum)

Размер зависит от того,  
насколько сложен скетч

Выбран неправильный  
последовательный порт

```
Serial port '/dev/tty.usbserial-A4001qa8' not found. Did you select the
java.awt.EventQueueThread.run(events(EventDispatchThread.java:170
)
    at
java.awt.EventQueueThread.run(EventDispatchThread.java:110)
```

Выбрана неправильная  
плата

```
Wrong microcontroller found. Did you select the right board from the T
Binary sketch size: 800 bytes (of a 7100 byte maximum)

avrdude: Expected signature for ATMEGA8 is 1E 93 07
Double check chip, or use -F to override this check.
```

жутко заумные сообщения об ошибке

# Возможные проблемы

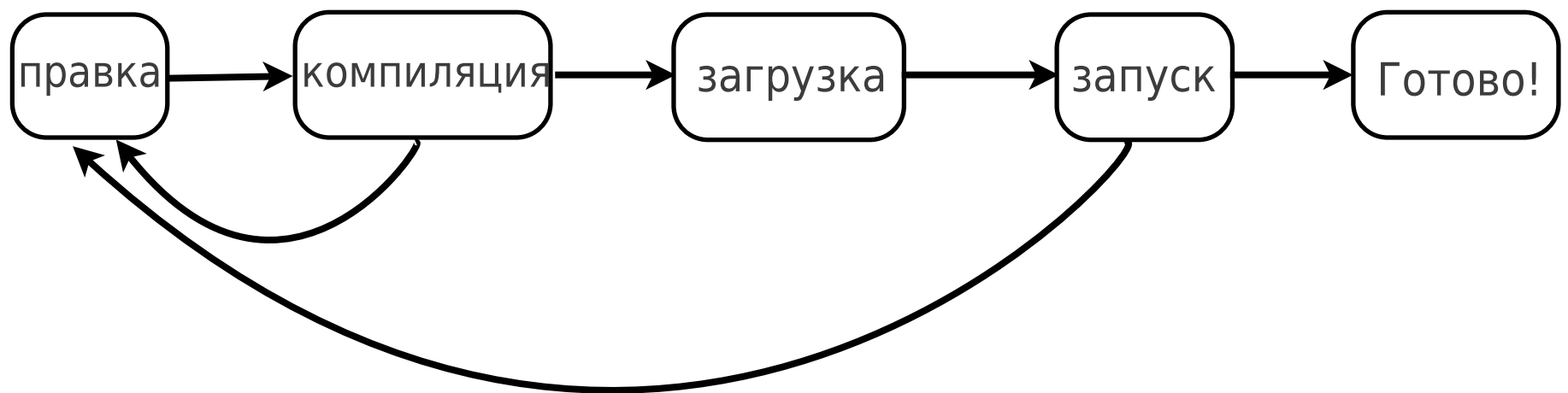
- Чаще всего проблема в неправильной настройке последовательного порта
- Если появляются "странные" сообщения об ошибке, просто попробуйте заново.
- Красный текст внизу - отладочная информация
- Статусные сообщения поясняют, что не так

# Я заставил светодиод мигать, что дальше?

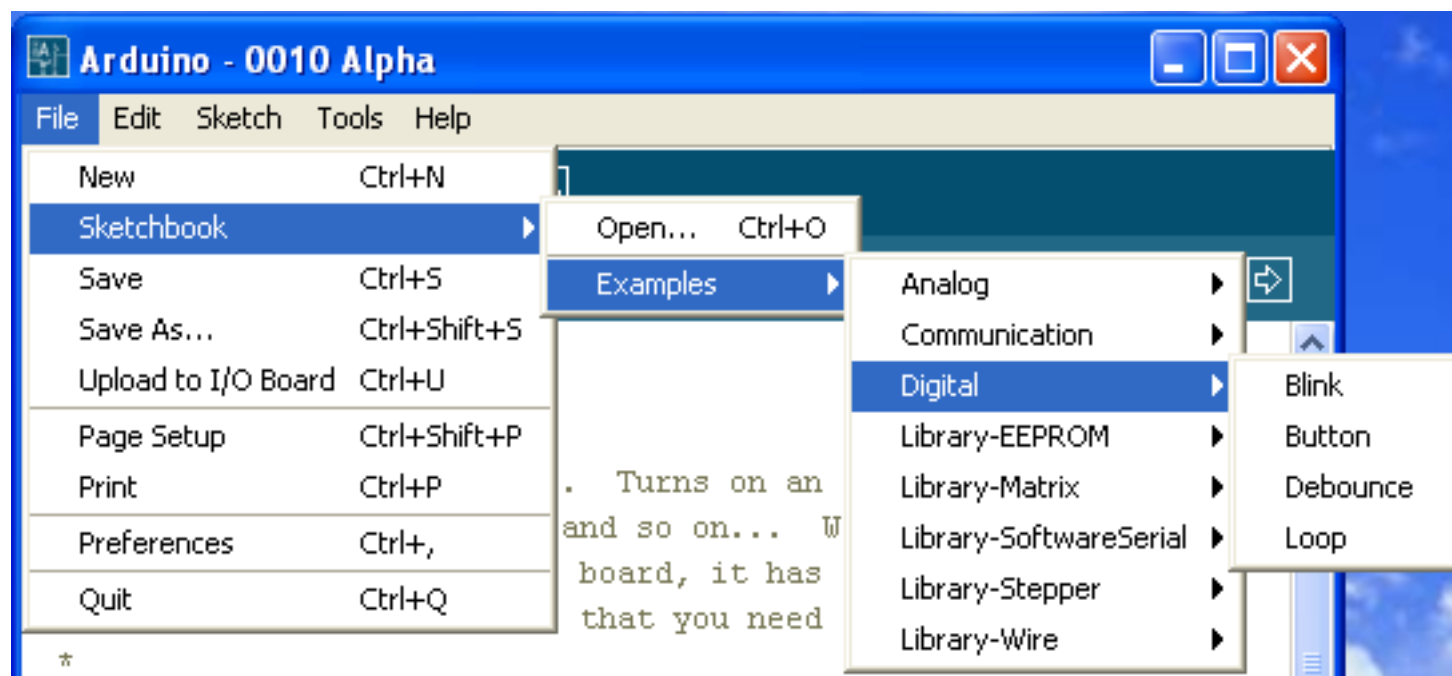
- Многие устройства включаются и выключаются цифровым сигналом
- Команда `digitalWrite()` позволяет управлять практически чем угодно
- Светодиоды - это просто, моторы будут дальше
- В Arduino есть 13 цифровых выходов, и легко можно добавить ещё при помощи специальных микросхем

# Цикл разработки

- Вносите сколько угодно изменений
- Не как в веб-программировании:  
правка → запуск
- Правка → компиляция → загрузка → запуск



# Множество встроенных примеров



Больше здесь:

<http://www.arduino.cc/en/Tutorial/HomePage>

И по всей Сети. Поищите "Arduino tutorial" или "Arduino notes" или что-то ещё, что Вам интересно, вместе со словом "Arduino", и скорее всего найдёте полезные страницы



# Перерыв

Возьмите кофе в Downbeat Cafe наверху

# «Язык» Arduino

- Язык - стандартный C (но упрощённый)
- Множество полезных функций
  - `pinMode( )` - сделать пин входом или выходом
  - `digitalWrite( )` - установить высокий/низкий уровень
  - `digitalRead( )` - прочитать состояние цифрового пина
  - `analogRead( )` - прочитать состояние аналогового пина
  - `analogWrite( )` - установить «аналоговое» значение
  - `delay( )` - подождать определённое время
  - `millis( )` - получить текущее время
- И многие другие. Плюс ещё в библиотеках

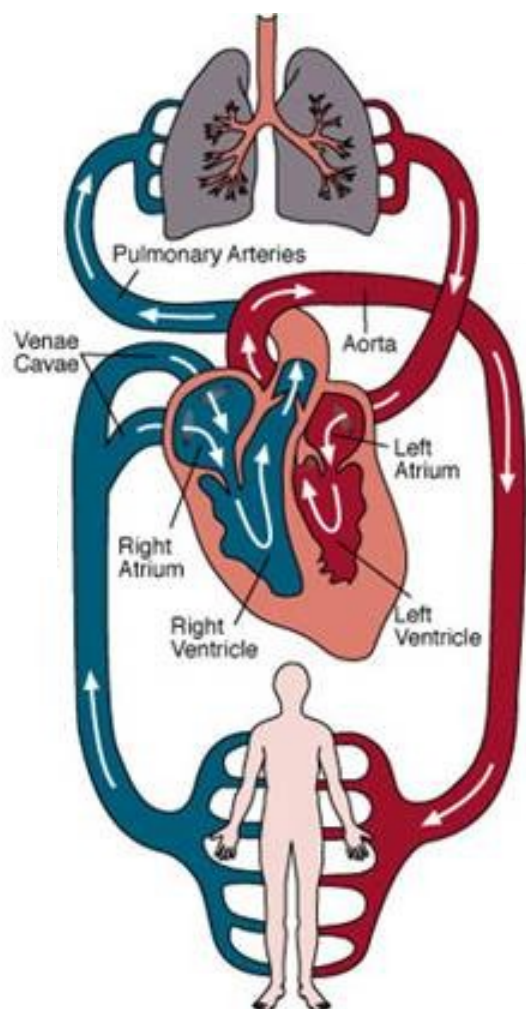
Также: библиотека работы с последовательным портом, с ЖК-экраном, с сервомашинкой...

# Структура скетча

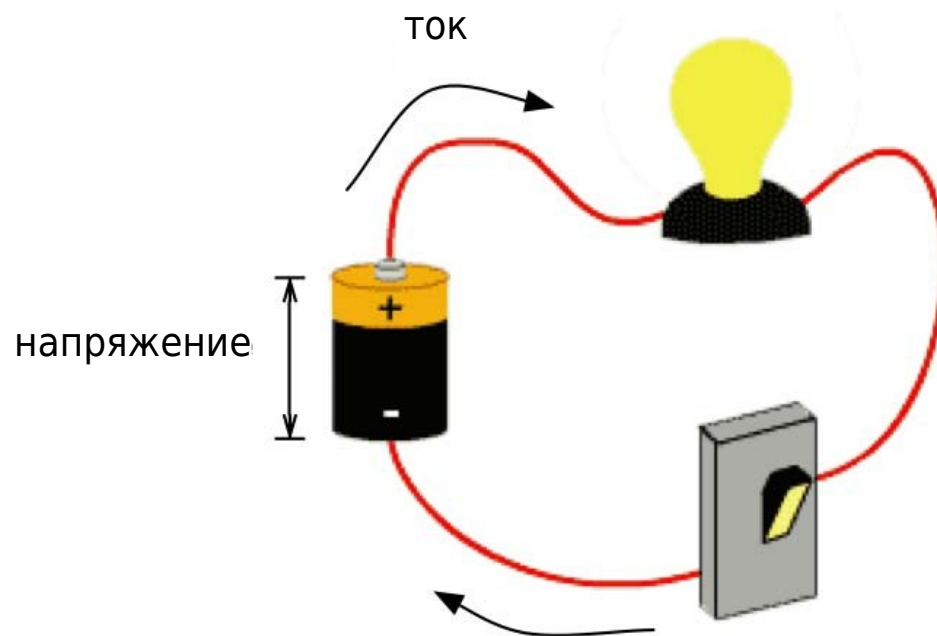
- Вверху - объявление переменных
- Инициализация
  - `setup( )` – запускается однажды в начале, установка пинов
- Запуск
  - `loop( )` - вызывается раз за разом после `setup( )`

Пины можно изменить и в `loop()`, но концептуально проще в `setup()`.

# Сборка схем



сердце бьётся,  
кровь течёт



напряжение толкает,  
ток течёт

Это всё к тому, что протекание тока - почти как течение жидкости.  
Некоторые электронные устройства задерживают течение тока, как маленькие  
вентили. Они называются «резисторы»

# Пример: светодиодный фонарик

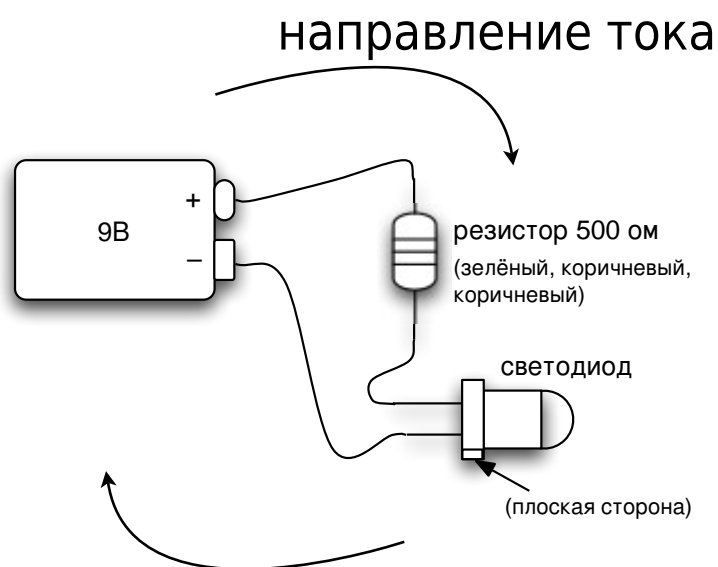
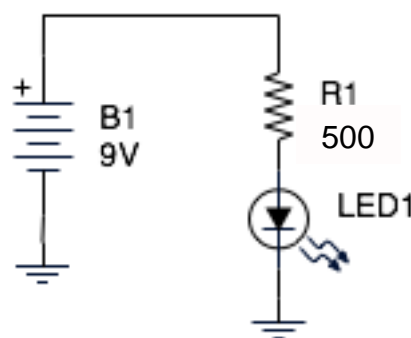
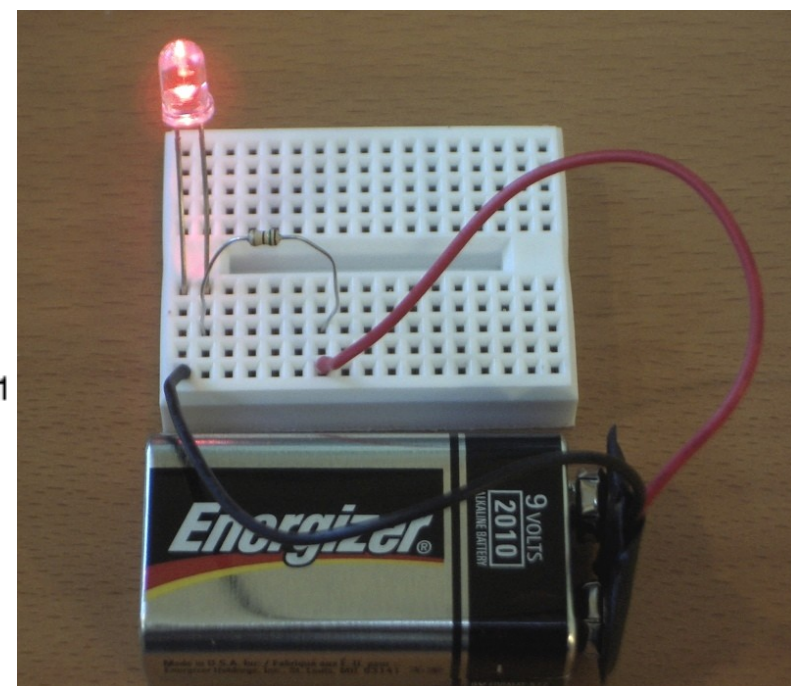


диаграмма  
соединений



принципиальная  
схема



собранная  
схема

Ток есть в замкнутой цепи. Его можно остановить,  
разорвав цепь

Все схемы со светодиодом такие: источник питания, ограничитель тока, светодиод.  
Плоская сторона светодиода - подключается к отрицательному полюсу схемы.  
Соответствует полоске на принципиальной схеме.  
Чем больше сопротивление - тем тусклее свечение. Чем меньше - тем ярче.  
Вам не нужно это собирать, но последующие схемы все очень похожи.

# Схема мигания светодиодом

«hello world» в мире микроконтроллеров

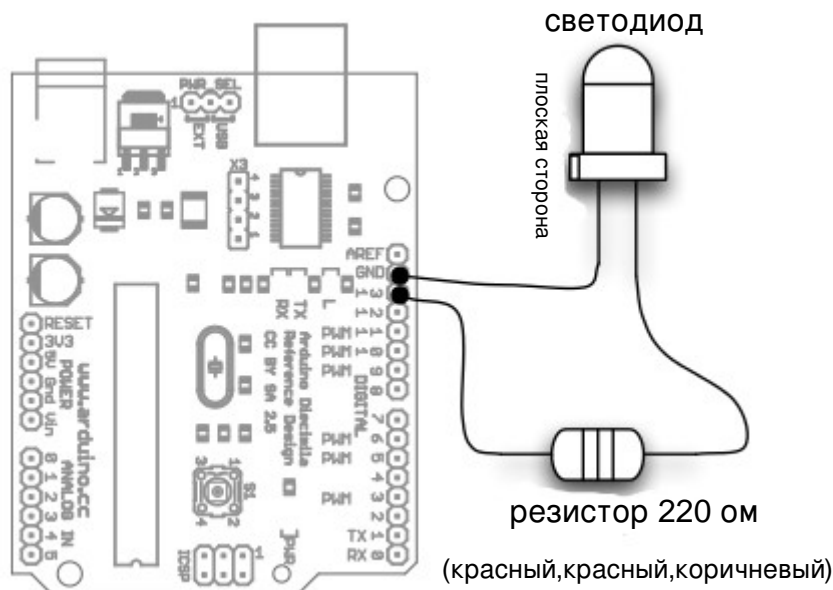
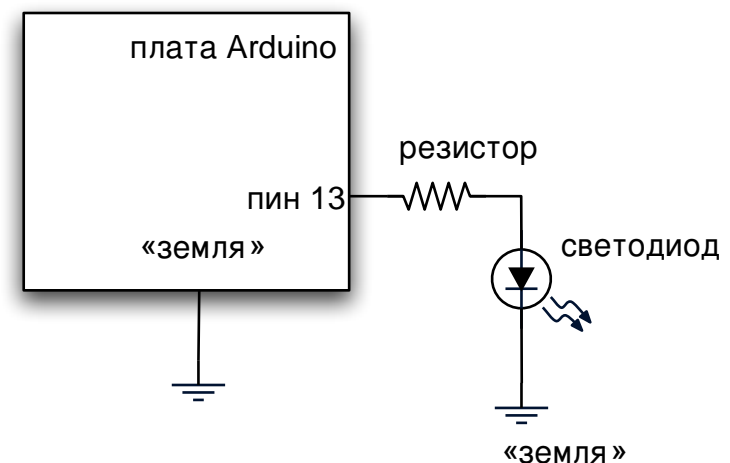


диаграмма  
соединений



принципиальная  
схема

В плате Arduino Diecimila эта схема уже встроена  
Чтобы включить светодиод: `digitalWrite(13, HIGH)`

Это - «управляемый компьютером светодиодный фонарик»

В схемах сигнал часто идёт из левого верхнего угла в правый нижний.

У часто встречающихся узлов, таких как «земля» (gnd), есть свои символы.

Вы можете собрать эту схему, используя любой цифровой пин, не важно какой.

Схема как на предыдущей странице, только «батарея» - это 13 пин Arduino,

и Вы можете включать и выключать светодиод.

Принципиальным схемам довольно легко научиться, немногие используют диаграммы соединений.

# Светодиоды и резисторы

У светодиодов полярность важна  
Короткий вывод - «минус», идёт на «землю»

светодиод

плоская сторона - "минус"

резистор

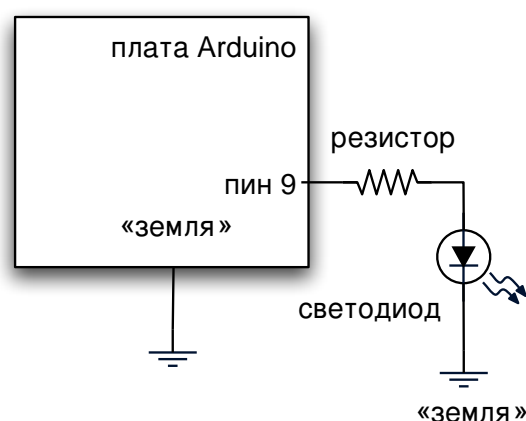
У резисторов полярность неважна





# Изменение яркости свечения

Схема, как в примере с мигающим, светодиодом, но пин 9 вместо 13



принципиальная  
схема

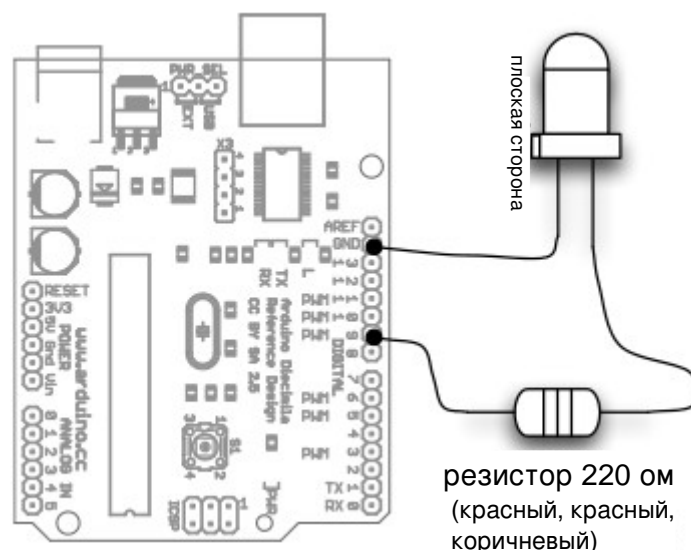
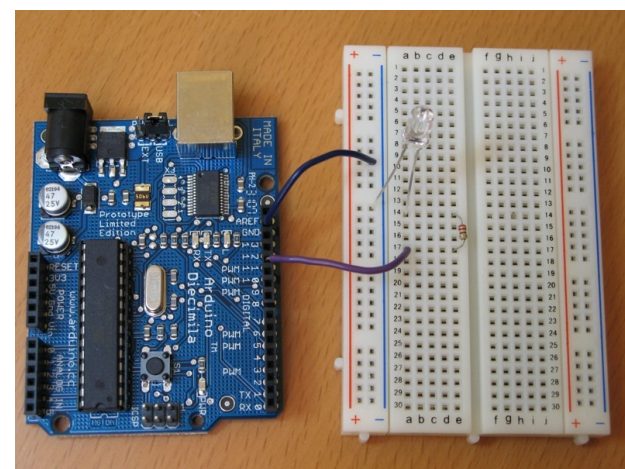


диаграмма  
соединений



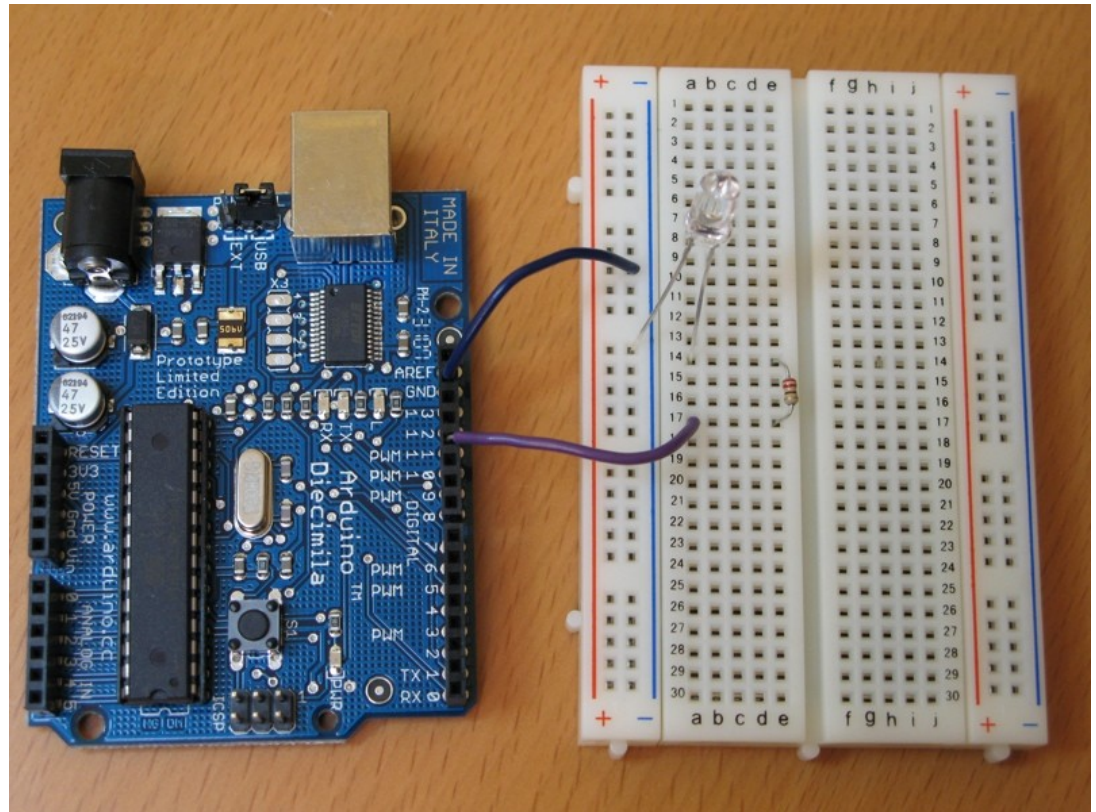
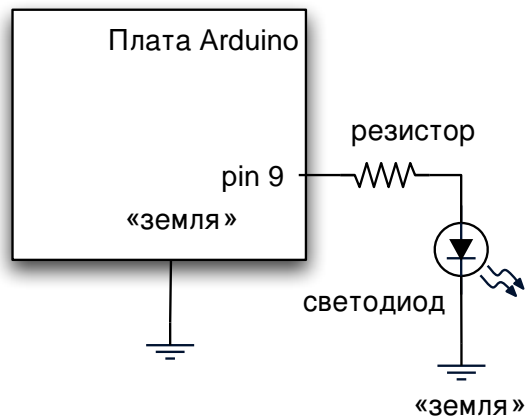
собранная  
схема

ШИМ-пины работают с командой `analogWrite(value)`  
где «value» - число от 0 до 255.

Чтобы включить светодиод с половинной яркостью: `analogWrite(9, 128)`

Дальше будет больше про ШИМ, но важно, что это работает только на пинах с надписью «PWM». Вкратце, это работает так: ток включается и выключается несколько сотен раз в секунду. Так что светодиод мигает, как в примере «Мигалка», но очень быстро. Наши глаза воспринимают его как более яркий или более тусклый. Нам нужно собрать эту схему.

# Так соберём её!



От абстрактной схемы переходим к реальному устройству

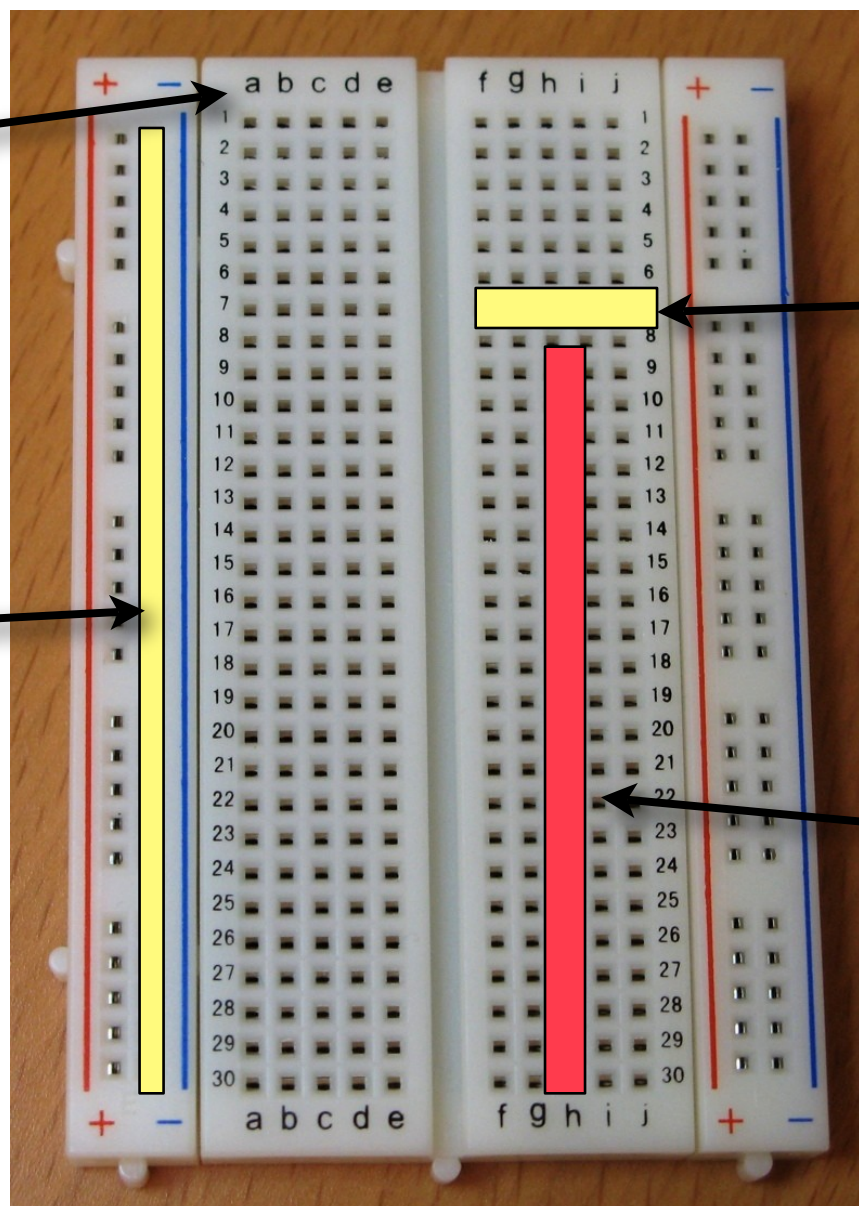
# Бесплаечные макетные платы

цифры и  
буквы — как  
координаты

Все соединены  
«общая шина»

соединены  
по 5 вместе

не  
соединены



Вставляйте провода в отверстия, чтобы делать соединения

\*Гораздо\* проще, быстрее, чем паять

Но: изнашиваются со временем; недешевы (изображённая на рисунке обошлась в 5\$, и это ещё повезло)



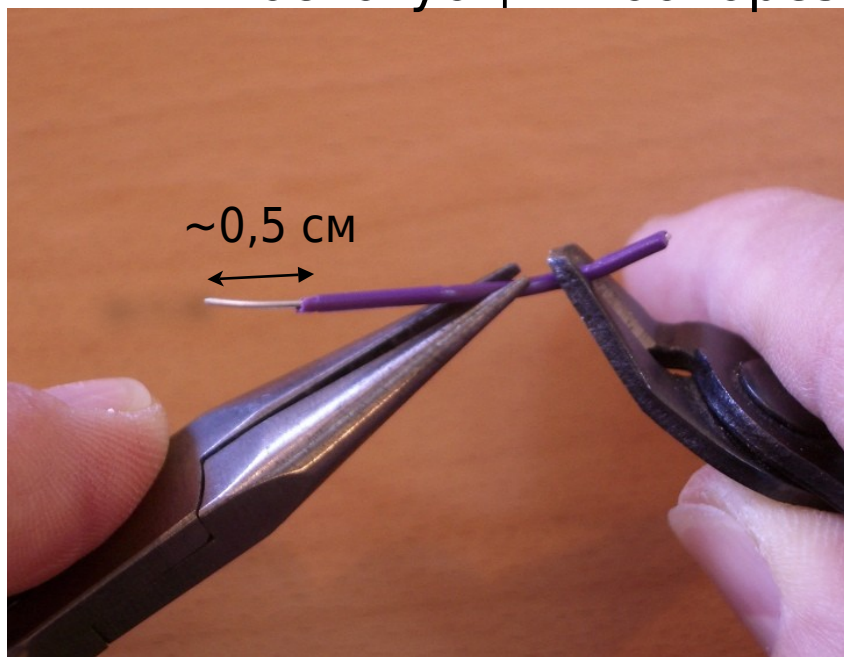
# Полезные инструменты



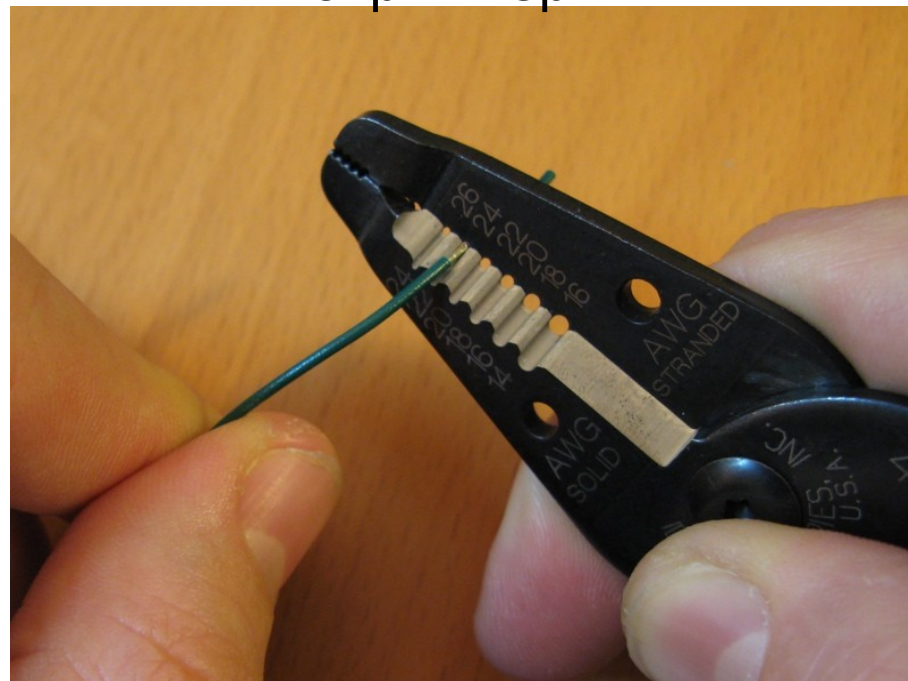
Даже с беспаячной макетной платой Вам придётся зачищать и обрезать провода. Каждый из этих инструментов стоит около 5\$. Если Вы можете позволить себе только один, возьмите зачистной инструмент (стриппер)

# Изготовление перемычек

плоскогубцы и бокорезы



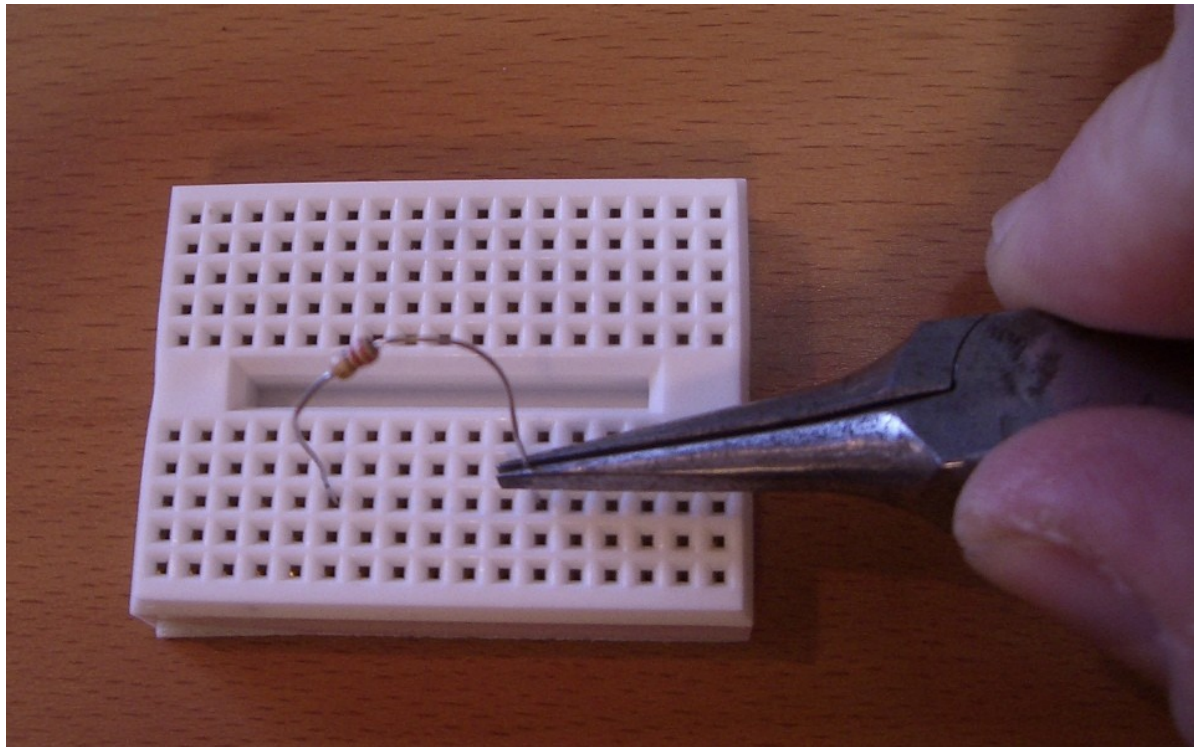
стриппер



Зачищайте примерно на полсантиметра  
И делайте перемычки той длины, какая нужна для Вашей схемы

# Работа с беспаячной макетной платой

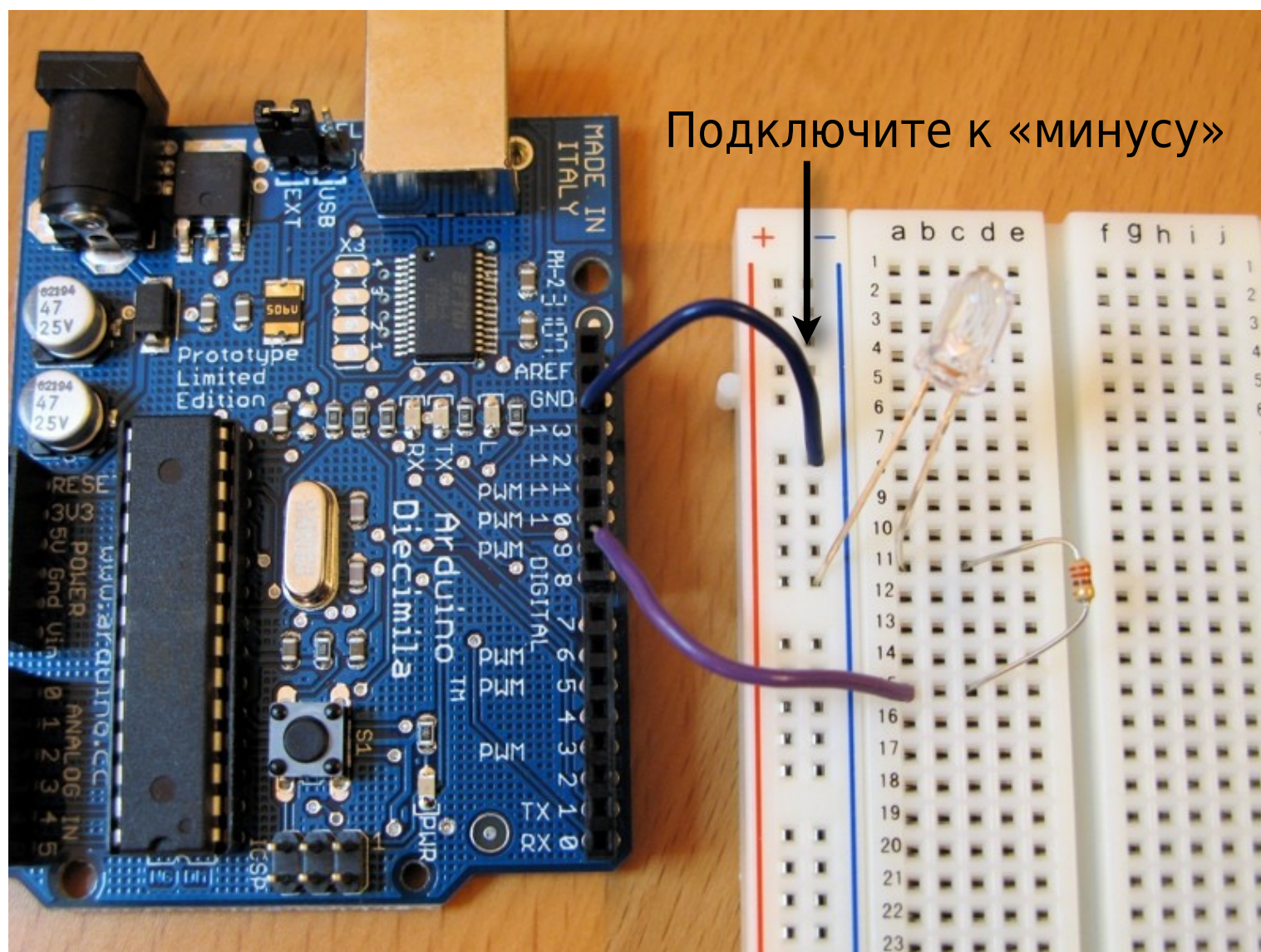
Плоскогубцы-утконосы помогают вставлять провода и компоненты в отверстия



Возьмите провод или вывод элемента за кончик и вставьте в отверстие



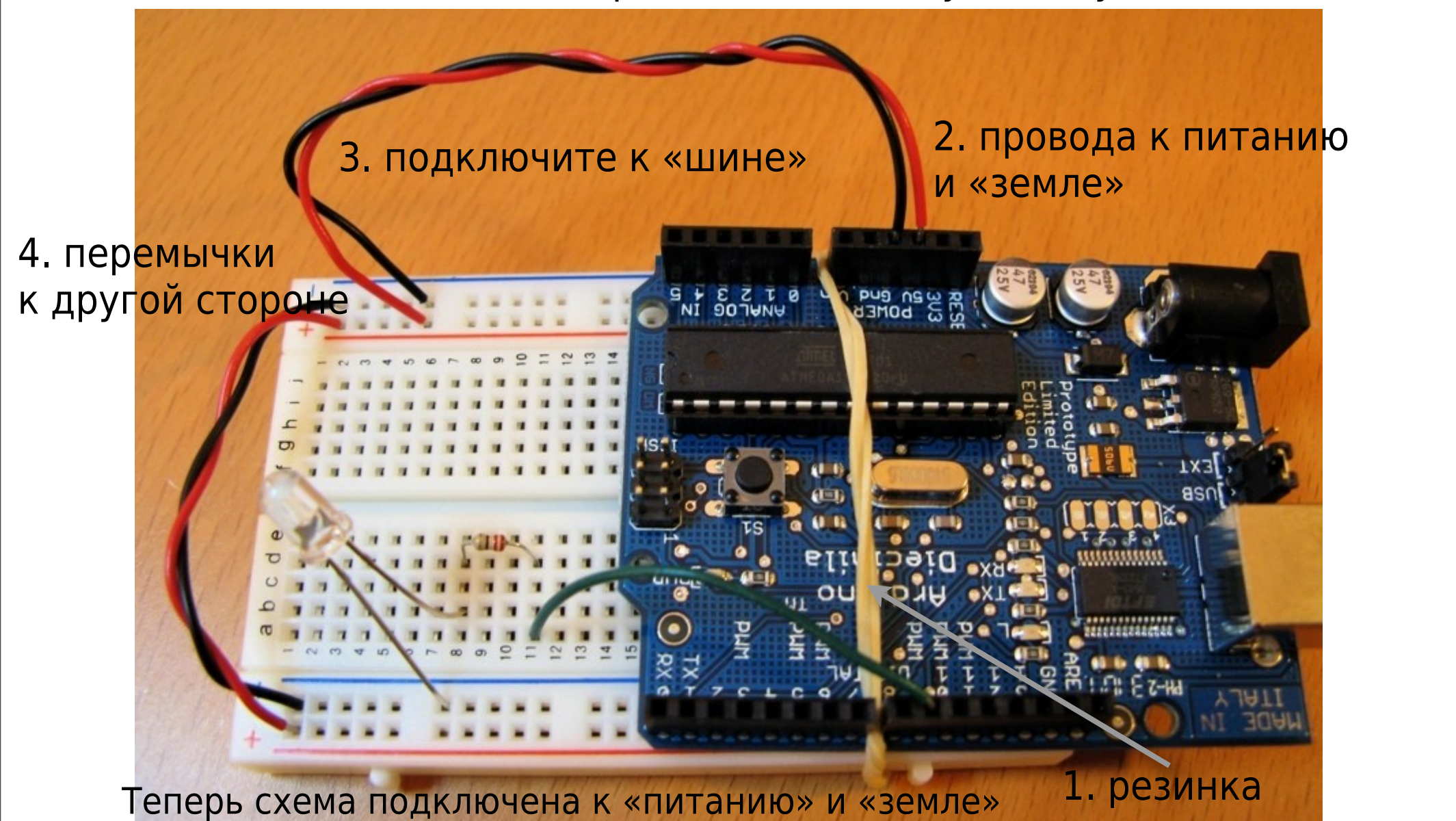
# Собранная схема





# Другой способ

Или: дешевый способ приделать макетную плату к Arduino



Так собирать схемы проще по двум причинам:

Во-первых, макетная плата и Arduino теперь тесно связаны, так что у проводов меньше шанс выпасть. Во-вторых, вы получаете множество отверстий для подключения питания и "земли", а обычно их нужно очень много.

Используйте эту конструкцию для остальных Ваших схем.

Трюк со скреплением Arduino и макетной платы резинкой бессовестно украден из курса Kimiko Ryokai "Осязаемый пользовательский интерфейс" (INFO290-13): <http://courses.ischool.berkeley.edu/i290-13/f07/>

# Скетч «Плавное включение»

Загрузите «File/Sketchbook/Examples/Analog/Fading»

ВНИМАНИЕ →

```
int value = 0;           // variable to keep the act
int ledpin = 9;          // light connected to digit

void setup()
{
  // nothing for setup
}

void loop()
{
  for(value = 0 ; value <= 255; value+=5) // fade in (from min to max)
  {
    analogWrite(ledpin, value);           // sets the value (range fr
    delay(30);                            // waits for 30 milli secor
  }
  for(value = 255; value >=0; value-=5)  // fade out (from max to mi
  {
    analogWrite(ledpin, value);
    delay(30);
  }
}
```

Нажмите «Upload». Через секунду светодиод будет плавно пульсировать

Уменьшите значения в «delay()», чтобы быстрее мигать

Попробуйте другие ШИМ-выводы (запомните: провода тоже нужно переставить)

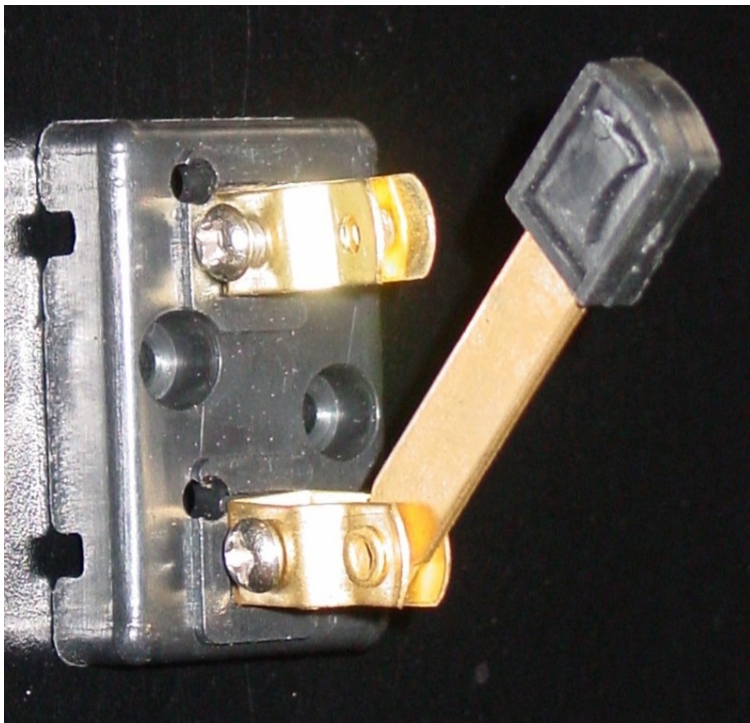
# Попробуйте с «Плавным включением»

- Сделайте его очень быстрым или очень медленным
- Переход от половинной на полную яркость
- Попробуйте другие ШИМ-пины
- Несколько плавно меняющих яркость светодиодов, с разной скоростью



# Датчики и Входы

Многие датчики - варианты переключателей  
Переключатели создают или разывают соединение



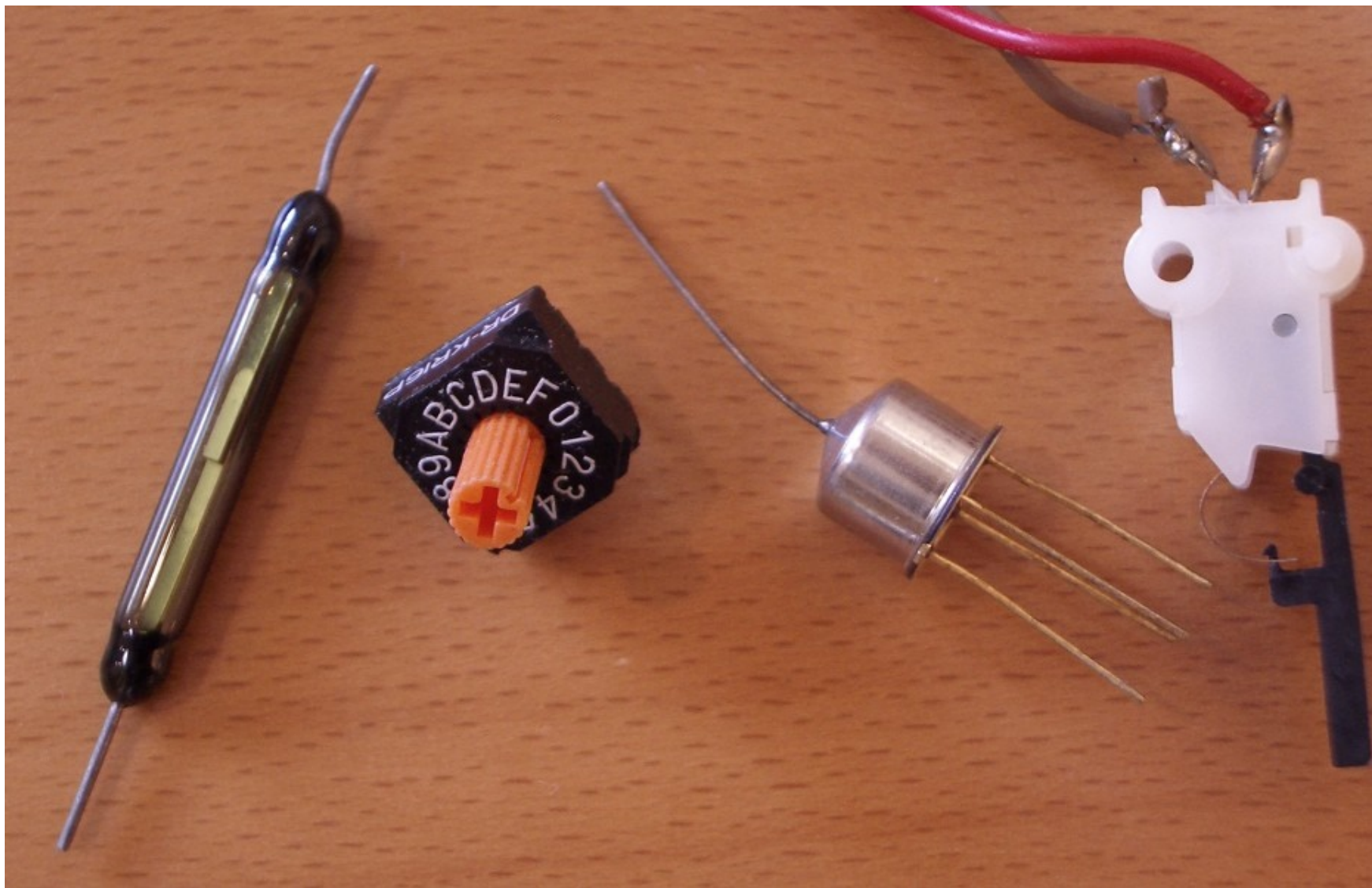
**рубильник**  
(SPST - Single-pole single-throw)



**тумблер**  
(SPDT — Single-pole double-throw)

В общем и целом, они все устроены как простые рубильники  
Однополюсный = управляет только одной цепью  
Двухполюсный = управляет двумя отдельными цепями одновременно  
Однонаправленный = только один путь для тока  
Двунаправленный = есть два пути для тока

# Разные типы переключателей



магнитный      шестнадцатеричный      датчик наклона      рычажный

В датчике наклона внутри маленький шарик, его можно услышать.  
Раньше они были ртутные, с настоящим шариком ртути внутри. Сейчас таких почти не делают.  
Магнитные переключатели (герконы) классные, но хрупкие  
Шестнадцатеричный состоит из нескольких переключателей и выдаёт 4 сигнала



# Самодельные переключатели

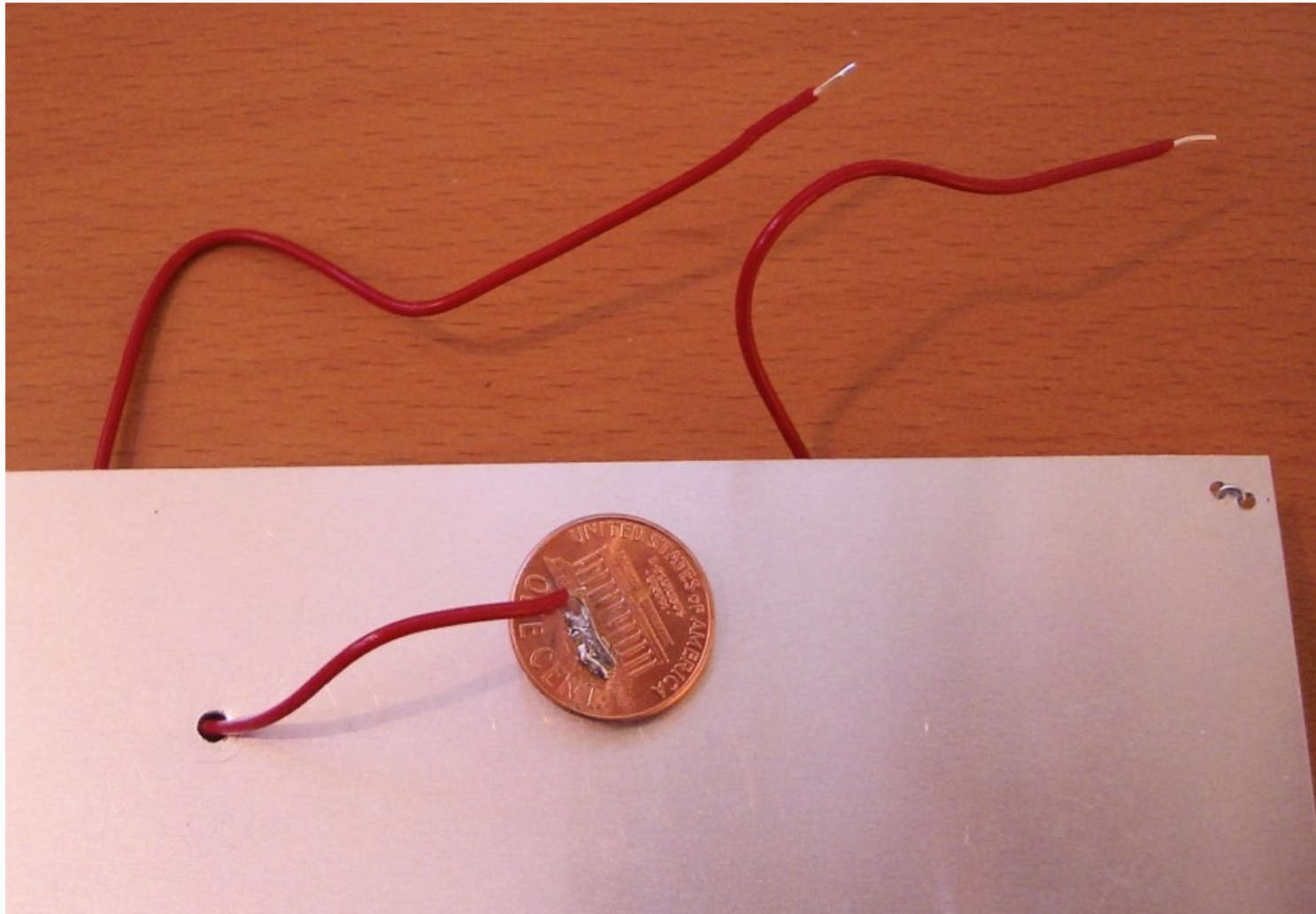
## «Хитрая монетка»

Монетка лежит на поверхности  
Если её поднимают, включается сигнализация



# Самодельные переключатели

## «Хитрая монетка»



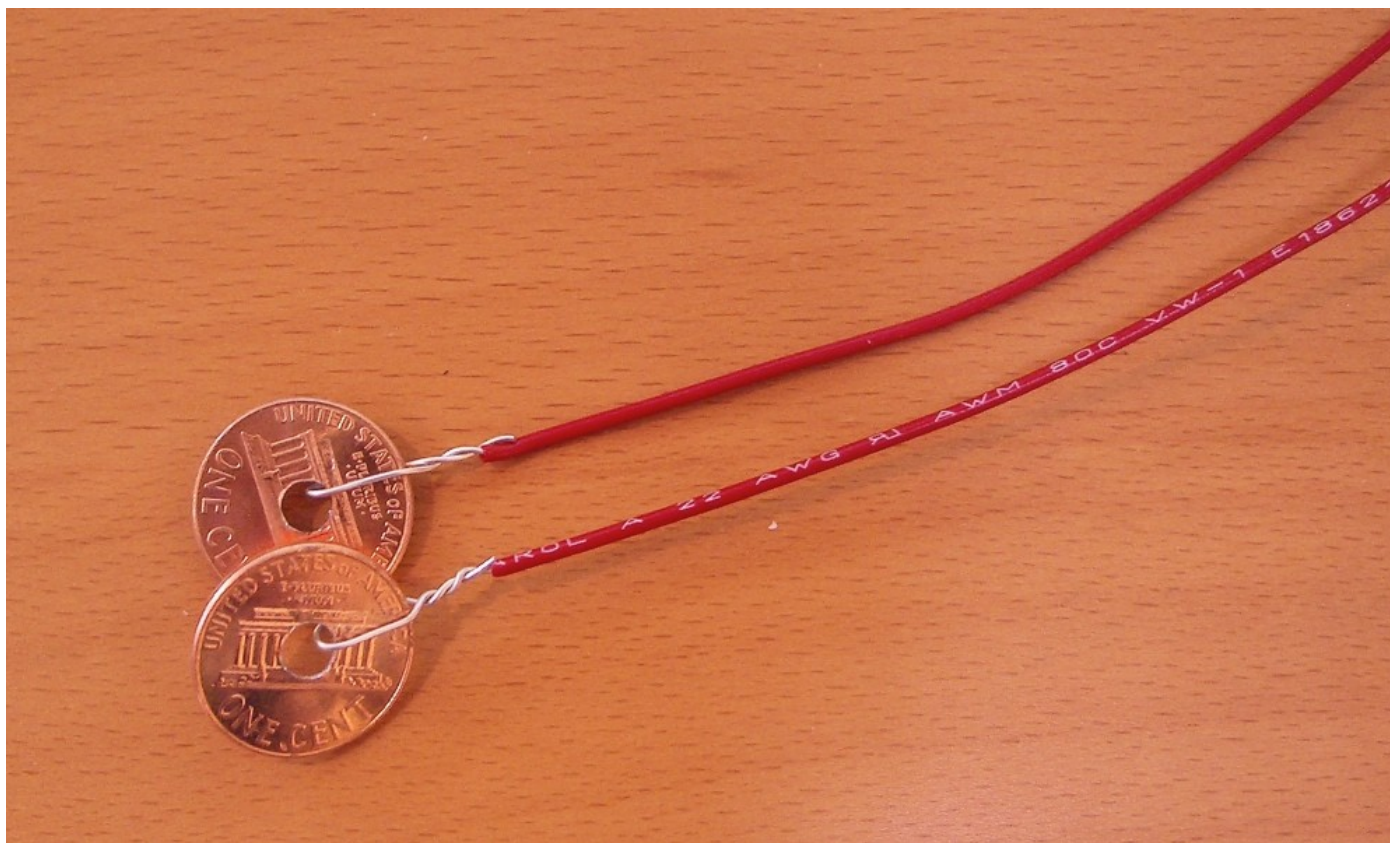
Поверхность — лист металла, проводящего электричество.  
Провод припаян к монетке.  
Провода припаяны или примотаны к листу металла.



# Самодельные переключатели

«Умные ветряные колокольчики»

Когда ветер дует достаточно сильно,  
Вам отсылается e-mail.

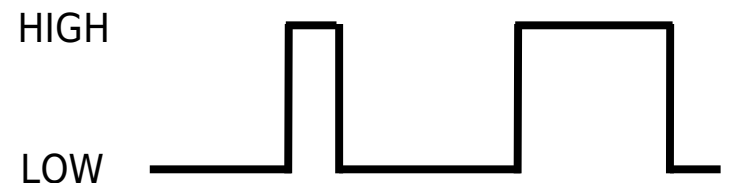


Следует использовать многожильный провод.

В коде анализируется последовательность импульсов вкл/выкл/вкл/выкл, чтобы определить наличие ветра

# Цифровой вход

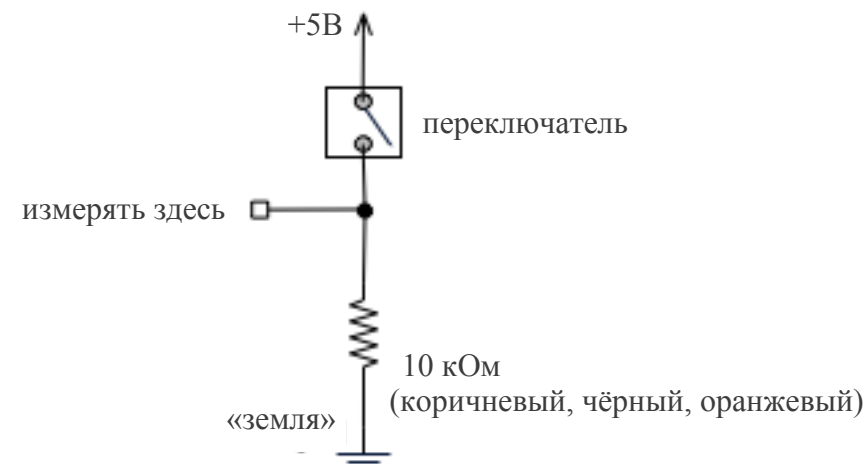
- Переключатели создают или разрывают соединение
- Но Arduino хочет знать напряжение
  - А именно, «HIGH» (5 вольт)
  - или «LOW» (0 вольт)



Как перейти от «соединить/разорвать»  
к «HIGH/LOW»?

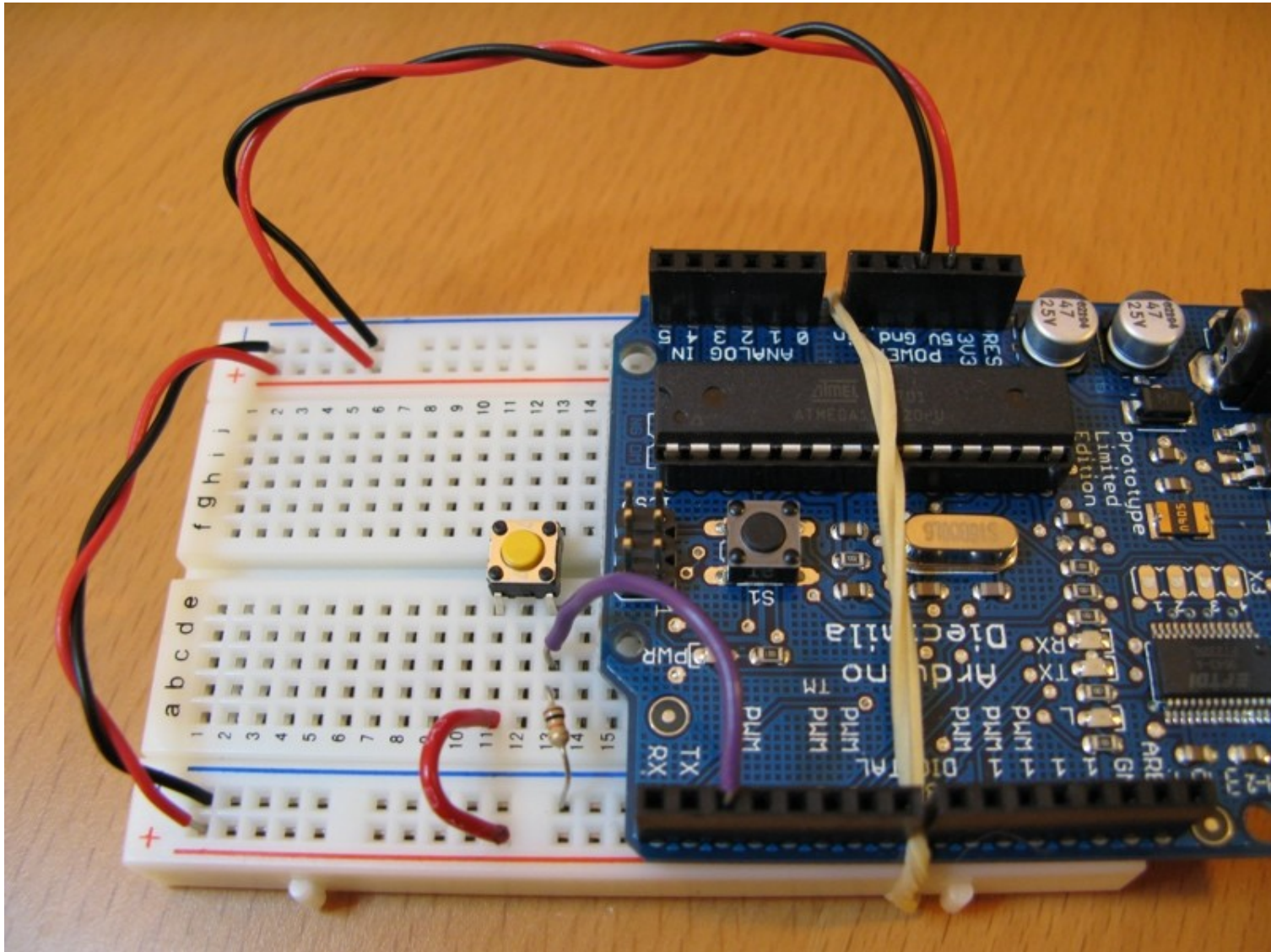
# От переключателя к HIGH / LOW

- Без подсоединений цифровые входы «плавают» между 0 и 5 вольтами (LOW и HIGH)
- Резистор «подтягивает» вход к «земле» (0 вольт)
- Нажатие выключателя «подтягивает» вход к 5V
- Нажато - HIGH  
Не нажато - LOW



Подтягивающий резистор не должен иметь маленькое сопротивление, иначе через него будет большой ток

# Соберём схему



Давайте подключим к пину 2

Вы можете оставить последний проект на плате, если хотите.

# Использование digitalRead()

- В `setup(): pinMode(myPin, INPUT)` делает ПИН ВХОДОМ
- В `loop(): digitalRead(myPin)` возвращает состояние переключателя
  - Если делаете много проверок, храните в переменной выходное значение `digitalRead()`.
  - **например** `val = digitalRead(myPin)`

# Скетч с цифровым входом

Загрузите «Sketchbook/Examples/Digital/Button»

```
int ledPin = 13;           // choose the pin for the LED
int inputPin = 2;          // choose the input pin (for a pushbutton)
int val = 0;               // variable for reading the pin status

void setup() {
  pinMode(ledPin, OUTPUT); // declare LED as output
  pinMode(inputPin, INPUT); // declare pushbutton as input
}

void loop(){
  val = digitalRead(inputPin); // read input value
  if (val == HIGH) {           // check if the input is HIGH
    digitalWrite(ledPin, LOW); // turn LED OFF
  } else {
    digitalWrite(ledPin, HIGH); // turn LED ON
  }
}
```

Теперь вы управляете миганием

(что нужно изменить, чтобы мигать внешним светодиодом,  
который вы подключили?)

Нажмите, чтобы выключить, отпустите, чтобы включить.

Обратите внимание, что мигает встроенный в плату светодиод.

Измените код, чтобы мигал светодиод, подключенный к пину 9.



# Используем выключатели, чтобы принимать решения

- Можно выбирать действия при помощи датчиков, работающих как переключатели
  - «Если есть человек, выстрелить из водяного пистолета»
  - «Если почва в горшке высохла, включить брызгалку»
- Определите действия, выберите их исходя из сигналов на входах
- Попробуем с теми действиями, которые уже знаем.



# FadeOrBlink

Загрузите скетч «FadeOrBlink» из раздатки

Схема, как в скетче  
«Плавное включение»

Комбинирует «Мигалку»  
и «Плавное включение» -  
вариант выбирается  
кнопкой

```
int ledPin = 9;           // choose the pin for the LED
int inputPin = 2;         // choose the input pin (for a pushbutton)
int val = 0;              // variable for reading the pin status
int fadeval = 0;          // variable for fading the LED

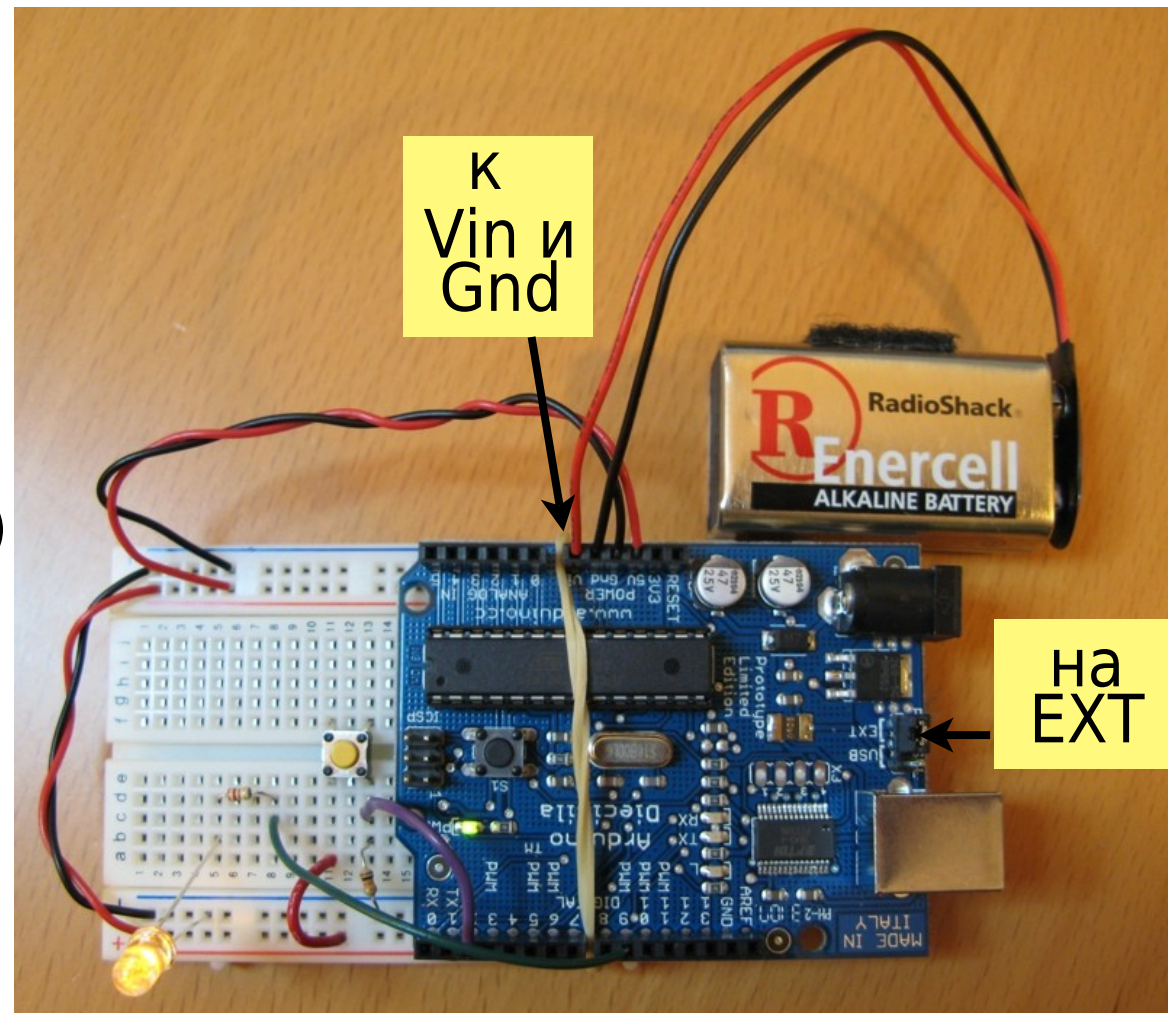
void setup() {
  pinMode(ledPin, OUTPUT); // declare LED as output
  pinMode(inputPin, INPUT); // declare pushbutton as input
}

void loop(){
  val = digitalRead(inputPin); // read input value
  if (val == HIGH) {           // pushed button means do blinking
    digitalWrite(ledPin, LOW); // turn LED OFF
    delay(50);
    digitalWrite(ledPin, HIGH); // turn LED ON
    delay(50);
  }
  else { // else button isn't pressed so do fading
    for(fadeval = 0 ; fadeval <= 255; fadeval+=5) { // fade in (from 0 to 255)
      analogWrite(ledPin, fadeval); // sets the value (range from 0 to 255)
      delay(10);
    }
    for(fadeval = 255; fadeval >=0; fadeval-=5) { // fade out (from 255 to 0)
      analogWrite(ledPin, fadeval);
      delay(10);
    }
  }
}
```

# Питание от батарейки

Arduino может работать полностью автономно. Это легко.

- Вначале загрузите скетч в Arduino
- Отключите кабель USB
- Переставьте перемычку с USB на EXT
- Подключите питание (7-12В постоянного тока)
- Индикатор питания светится. Работает!
- Прodelайте всё в обратном порядке, чтобы перепрограммировать



# Питание от батарейки

- Подключаться к гнездам не очень удобно
- Лучше использовать разъём питания
- Работает хорошо, но требует немного пайки



# Что дальше

- Сделайте свои выключатели: алюминиевая фольга, монетки и т.д.
- Соберите бегущий огонёк или стробоскоп
- Соберите фонарик для велосипеда, который светится, только когда жмёте на педали
- Сделайте Раздражающую Мигалку™  
(сверк-сверк-сверк, стоп.... сверк-сверк-сверк)

Множество вещей, которые можно сделать только с переключателями и светодиодами

# КОНЕЦ занятия 1

<http://todbot.com/blog/bioniscarduino/>

Тод Е. Курт

[tod@todbot.com](mailto:tod@todbot.com)

Не стесняйтесь писать мне на почту, если есть вопросы.

# Полезные ссылки

<http://arduino.cc/>

Официальная страница. Также посмотрите «игровую площадку» (Playground) и форумы

<http://ladyada.net/learn/arduino/>

Отличные обучающие статьи по Arduino

<http://todbot.com/blog/category/arduino/>

Разные фильмы, хаки, руководства по Arduino

<http://freeduino.org/>

Каталог знаний по Arduino

<http://adafruit.com/>

Наборы для начинающих с Arduino, клон Arduino Boarduino, много классных наборов

<http://sparkfun.com/>

Продаёт платы Arduino и множество хороших датчиков и прочей периферии

Книги:

“Physical Computing”, Dan O’Sullivan & Tom Igoe

“Making Things Talk”, Tom Igoe

“Hacking Roomba”, Tod E. Kurt

Обязательные ссылки на литературу