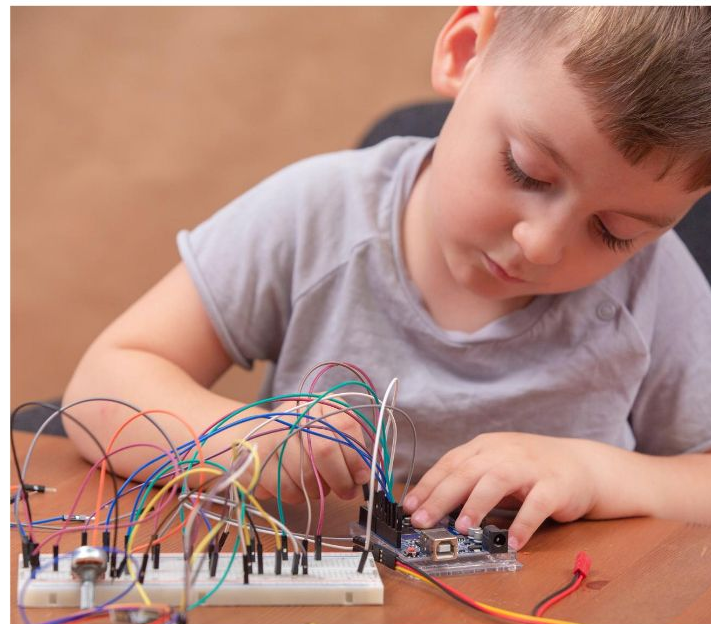
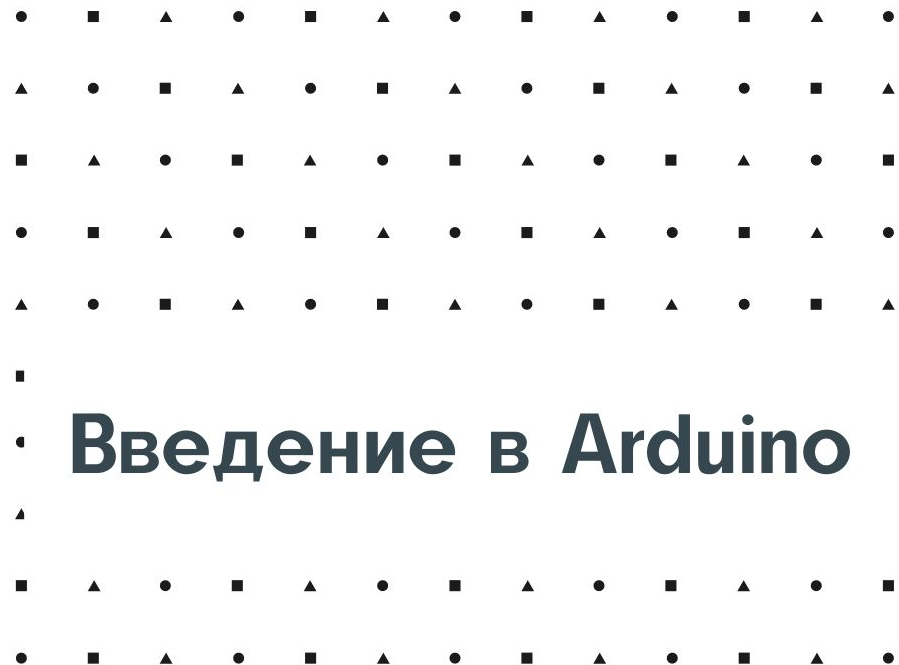




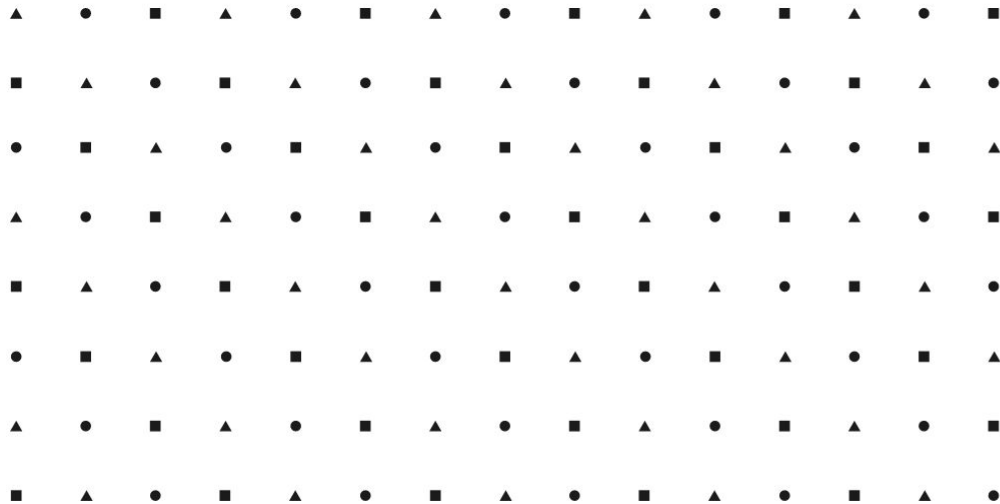
Enjoy Robotics

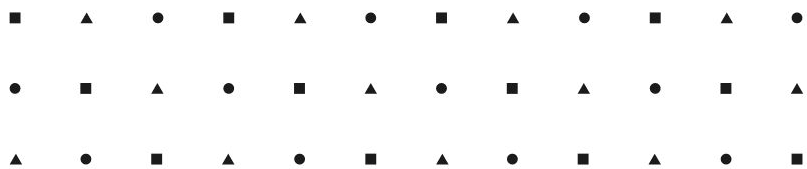


Двигаясь по этой инструкции вы научитесь собирать схемы подключения светодиода, пищалки, датчика, сервопривода и даже создадите настоящий парковочный радар. Нас ждет погружение в мир схемотехники, программирования и Arduino

Если вы хотите сразу начать собирать и программировать робота Отто, то переходите к второму шагу на сайте

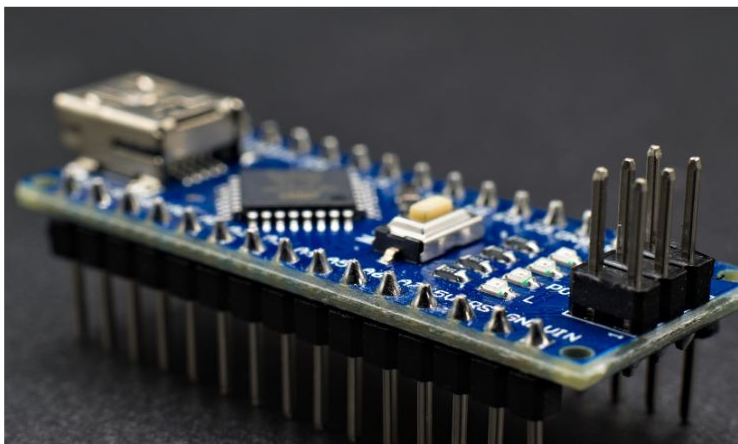
<https://enjoy-robotics.ru/otto-1>





# Arduino Nano

## Программно-аппаратный модуль



Эта отладочная плата небольшого размера, которая входит в тройку лидеров по популярности среди радиолюбителей-программистов. Несмотря на свой скромный размер, она практически ничем не уступает нашумевшей Arduino Uno по функционалу и может использоваться в малогабаритных проектах

Плата позволяет использовать 14 цифровых контактов ввода/вывода. Их обозначение начинается с латинской буквы «D», от английского слова digital-цифровой. Каждый из этих контактов может быть настроен как на вход, так и на выход цифрового сигнала

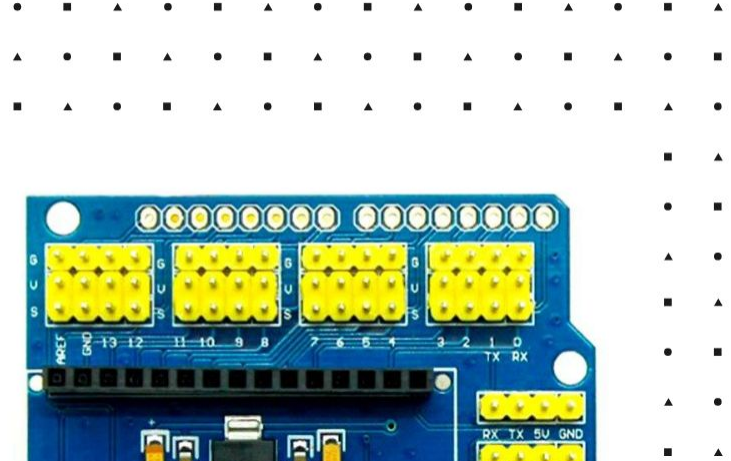
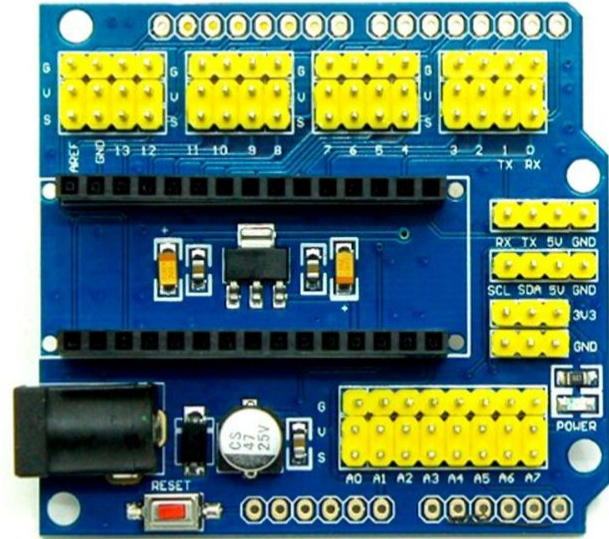
Помимо цифровых контактов, плата Arduino Nano имеет на своём борту 8 аналоговых входов, обозначенных диапазоном A0-A7



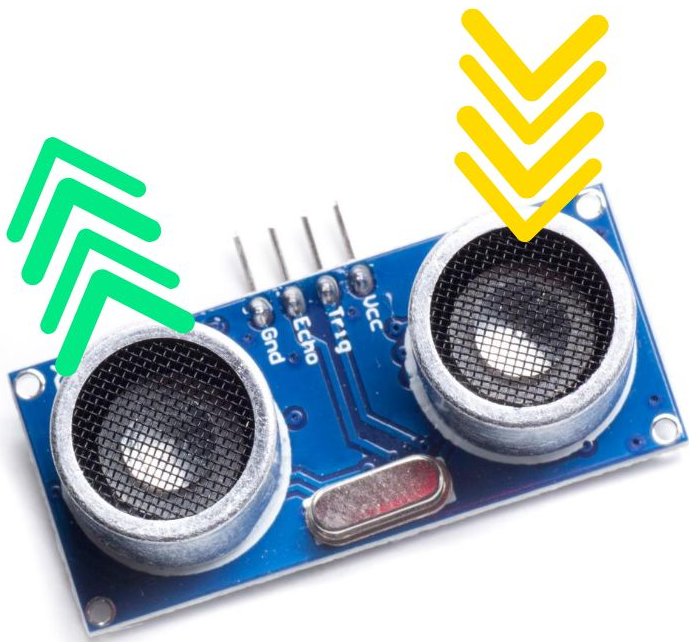
# Плата расширения Arduino nano I/O

Плата расширения Arduino – это законченное устройство, предназначенное для подключения датчиков и модулей к основному контроллеру с помощью стандартных разъемов. Такие платы, совершенно логично называемые платами расширения, служат для выполнения самых разнообразных задач и могут существенно упростить жизнь ардуинщика. Зачем тратить время, чтобы проектировать, припаивать и отлаживать то, что можно взять уже в собранном виде, и сразу начать использовать?

Шилд Arduino Nano I/O идеально подходит для работы со всеми электронными компонентами из набора. Модуль предназначен для увеличения количества контактных групп подключаемых модулей класса Arduino и им подобных. Одно из важных преимуществ - на разъемы кроме информационных сигналов выведены линии питания (V) и линии земли (G)



# Ультразвуковой дальномер HC-SR04



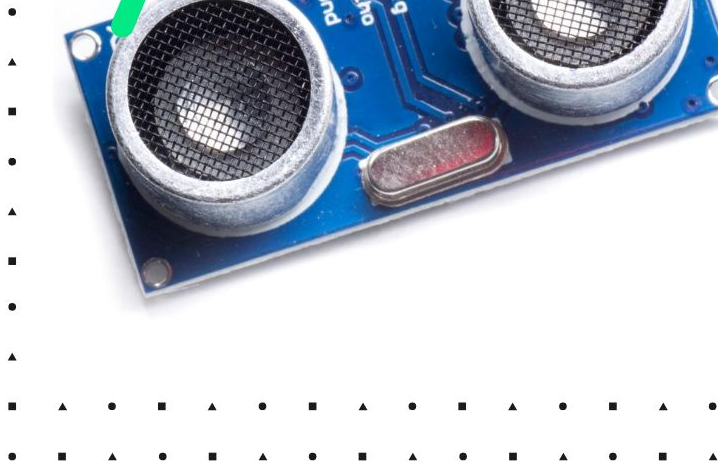
## Принцип сонара

Этот дальномер может служить прекрасным датчиком для робота, благодаря которому он сможет определять расстояния до объектов, объезжать препятствия или строить карту помещения. Его можно также использовать в качестве датчика для сигнализации, срабатывающего при приближении объектов

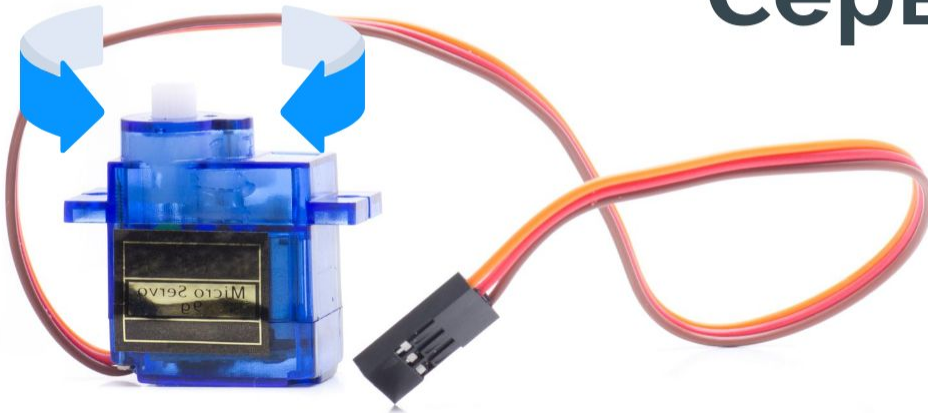
Сенсор работает как сонар: посылает ультразвуковой пучок и по задержке отражённого сигнала определяет расстояние до цели. Так же в природе ориентируются летучие мыши

Он генерирует звуковые импульсы на частоте 40 кГц и слушает эхо. По времени распространения звуковой волны туда и обратно можно однозначно определить расстояние до объекта

- Vcc — положительный контакт питания
- Trig — цифровой вход. Для запуска измерения необходимо подать на этот вход логическую единицу
- Echo — цифровой выход. После завершения измерения, на этот выход будет подана логическая единица на время, пропорциональное расстоянию до объекта.
- GND — отрицательный контакт питания



# Сервопривод SG-90

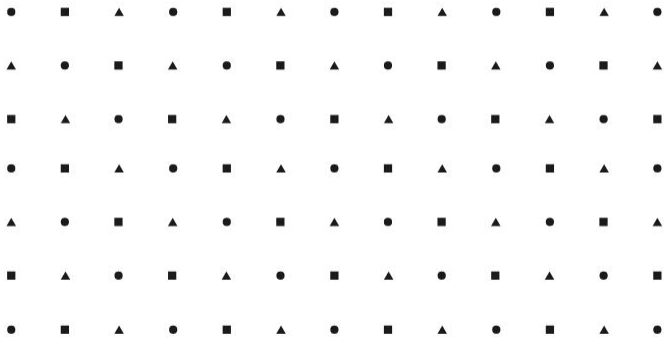


## Главный механизм

Под сервоприводом чаще всего понимают механизм с электромотором, который можно попросить повернуться в заданный угол и удерживать это положение. Однако, это не совсем полное определение

Сервоприводом является любой тип механического привода, имеющий в составе датчик (положения, скорости, усилия и т.п.) и блок управления приводом, автоматически поддерживающий необходимые параметры на датчике и устройстве согласно заданному внешнему значению

Включая и выключая электромотор, можно вращать выходной вал — конечную шестерню сервопривода, к которой можно прикрепить что-то, чем мы хотим управлять



# Пьезодинамик

## Звуковые сигналы робота



Зуммер, пьезопищалка – все это названия одного устройства. Данные модули используются для звукового оповещения в тех устройствах и системах, для функционирования которых в обязательном порядке нужен звуковой сигнал. Зуммеры используют в различной бытовой технике и игрушках, использующих электронные платы. Пьезопищалки преобразуют команды, основанные на двухбитной системе счисления 1 и 0, в звуковые сигналы.

Принцип действия зуммера основан на открытом братьями Кюри в конце девятнадцатого века пьезоэлектрическом эффекте. Согласно ему, при подаче электричества на зуммер он начинает деформироваться. При этом происходят удары о металлическую пластинку, которая и производит “шум” нужной частоты.

Подключение модуля пьезоэлемента к Ардуино выглядит достаточно простым. Потребляемый им ток маленький, поэтому его можно просто напрямую соединить с нужным контактом.

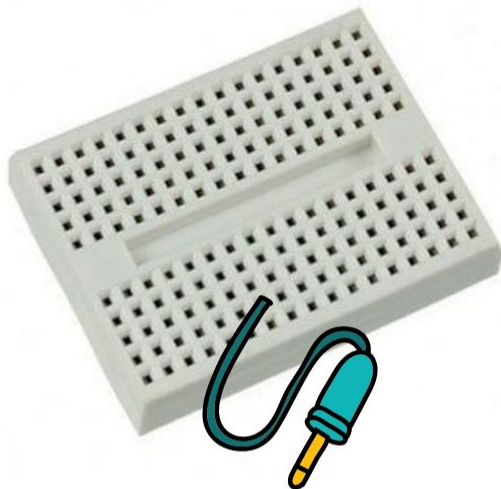
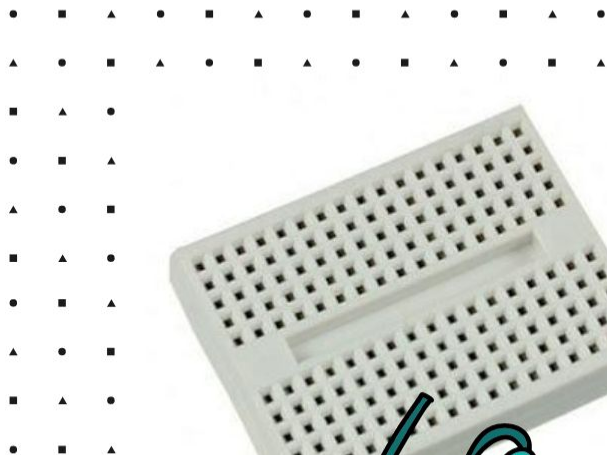


# Макетная плата

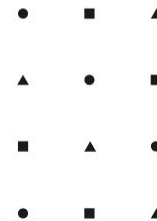
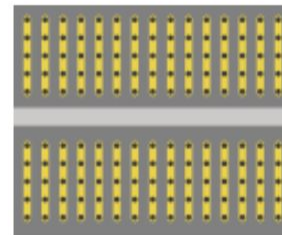
## Беспаячная макетная плата на 170 точек

Так называемая breadboard или макетная плата — незаменимая вещь для экспериментов с электроникой. Она позволяет быстро, удобно, без паяльника собирать электрические схемы

При создании чего-то нового, в процессе проб и ошибок почти всегда приходится несколько раз корректировать схему. Если все детали сразу соединять пайкой, изменения становятся проблемой. Макета позволяет не думать об этом и вносить сколько угодно изменений



## Контактные площадки макетной платы



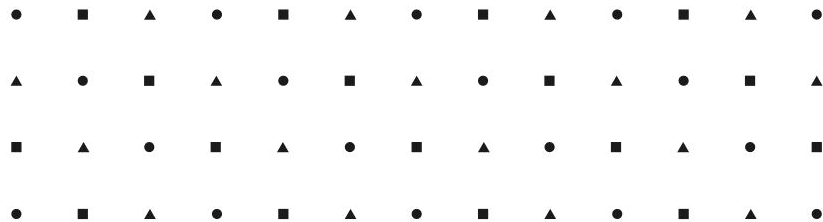
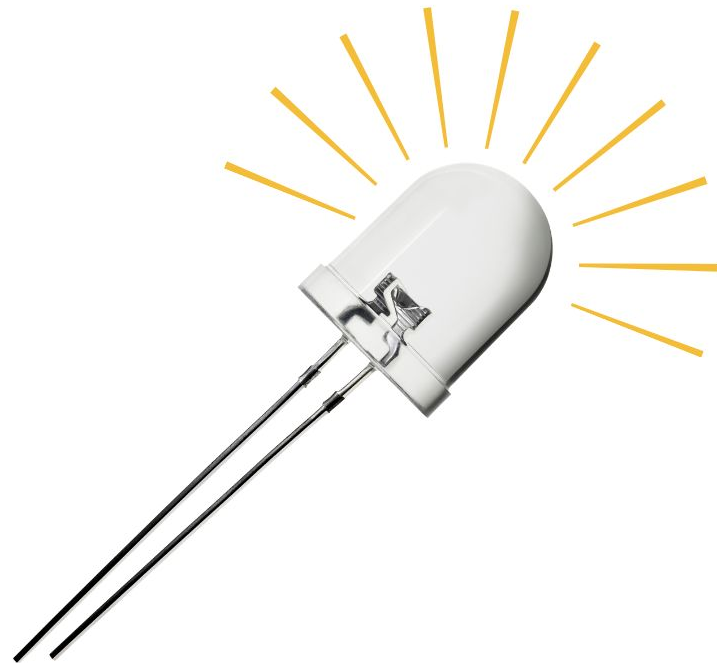


# Светодиод

## Световая индикация

Светодиод (англ. Light Emitting Diode или просто LED) — энергоэффективная, надёжная, долговечная «лампочка» Светодиод — вид диода, который светится, когда через него проходит ток от анода (+) к катоду (-).

Не трудно заметить, что светодиод имеет разные по длине металлические ножки, именно по ним определяется полярность при его подключении: длинная отвечает за тот самый сигнальный вход (+), а короткая будет подключаться к земле (GND) нашей Arduino Nano



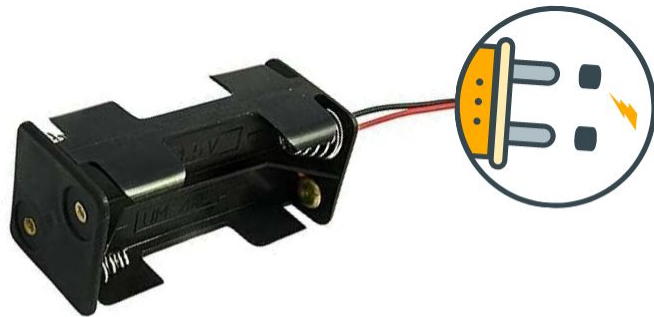
## Набор проводов

При прототипировании электронных систем на макетных платах с участием Arduino или какой-либо другой платформы невозможно обойтись без соединительных проводов, по которым будут передаваться сигналы, и протекать питание ко всем датчикам и модулям. Наши провода имеют на обоих концах прочные металлические штыри, которые очень удобно вставлять в отверстия макетных плат



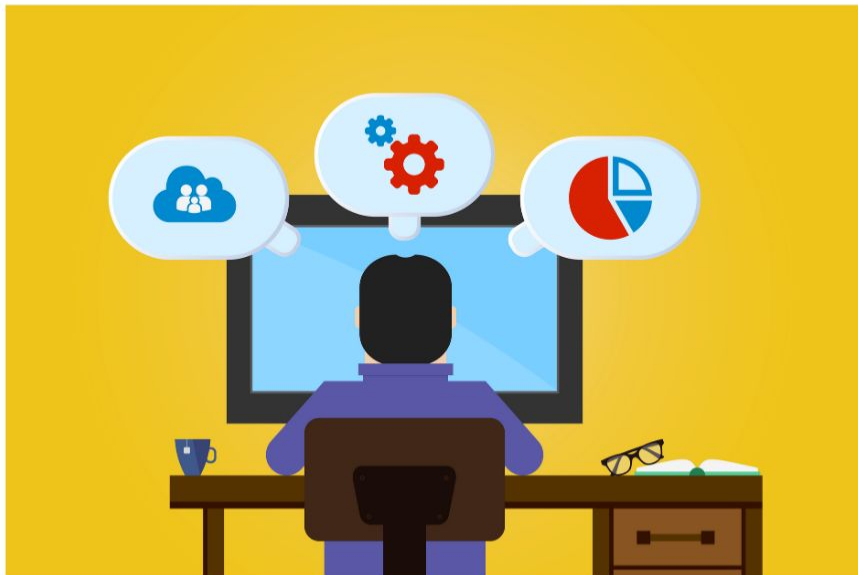
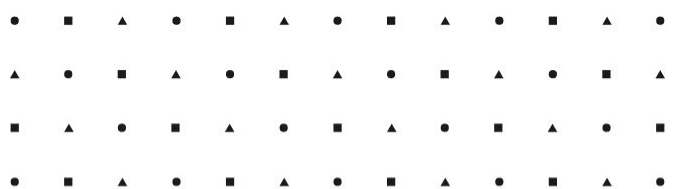
## Батарейный отсек

Батарейный отсек на четыре элемента AA - это практически готовый Power Bank для нашего робота. Этот батарейный бокс полностью прячет в своем корпусе 4 батарейки AA и имеет механический выключатель питания. Аккумуляторный блок можно использовать для автономного питания различных устройств на Ардуино, роботов, RC моделей и других устройств требуемых автономного батарейного питания



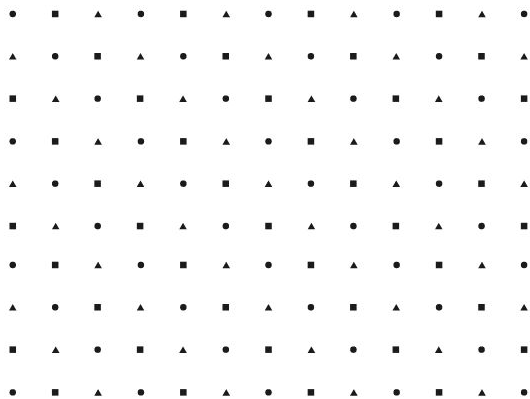
# Этапы обучения

## План работы



1. Установка mBlock
2. Сборка прототипов схем и программирование базовых проектов Arduino mBlock
3. Установка OttoBlokly
4. Сборка и калибровка робота Отто
5. Программирование робота Отто





## Блоковая среда программирования

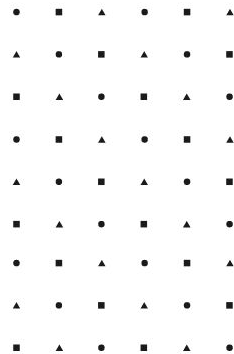
Сегодня она активно используется как для MakeBlock, так и для самого широкого спектра плат Ардуино. Программа русифицирована и базируется на Scratch 3, позволяя с ее помощью создавать собственных роботов и технические проекты произвольной конфигурации на Ардуино

## Преимущества mBlock

- поддержка большинства плат Ардуино
- возможность использования собственных блоков с возможностями рекурсии
- простота использования библиотек и наработок в собственных проектах
- удобство создания своих программ; можно изучать синтаксис языка программирования на базе mBlock



mBlock 5




# Установка mBlock

Ссылка для скачивания и установки:

<https://www.mblock.cc/en-us/download>


## Download mBlock

One-stop coding platform tailored to coding education,  
trusted by 15 million educators, and learners



**mBlock web version**  
Chrome browser recommended >>  
Support Windows/Mac/Linux/Chromebook


[Code with blocks](#) [Code with Python](#)




**mBlock PC version**  
Version: V5.2.0  
Released: 2020.01.22  
[Released log >>](#) [Previous version >>](#)

[Download for Windows](#) [Download for Mac](#)


Win7 or Win10 (64-bit recommended) macOS 10.12+



**mBlock mobile app**  
Learn coding in phones and tablets



**Android**  
Android 5.0 +  
(ARM-based devices only, X86  
Android not supported)

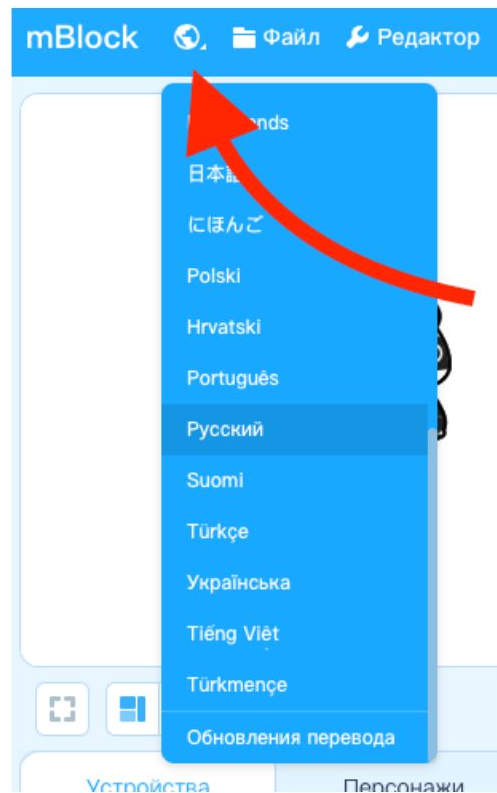


**iOS**  
iOS 10.0 +

# Шаг 1

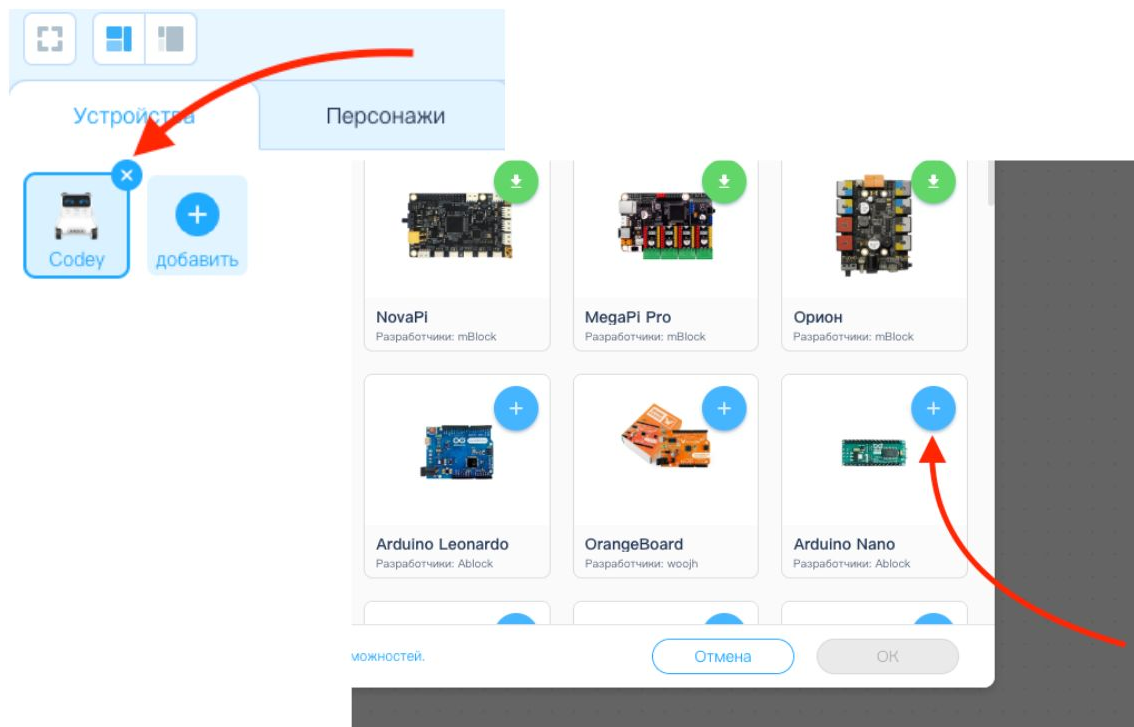
## Настройка Mblock

Выберем русский язык



# Программирование Arduino

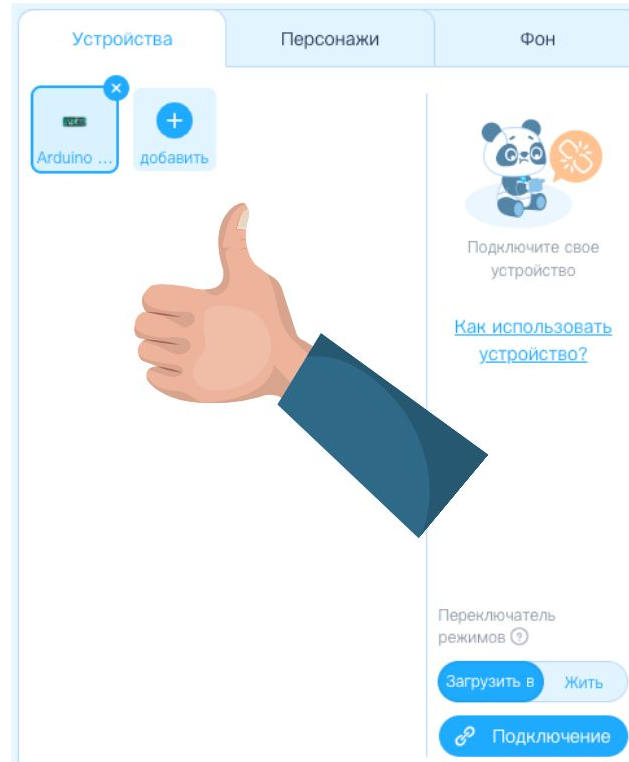
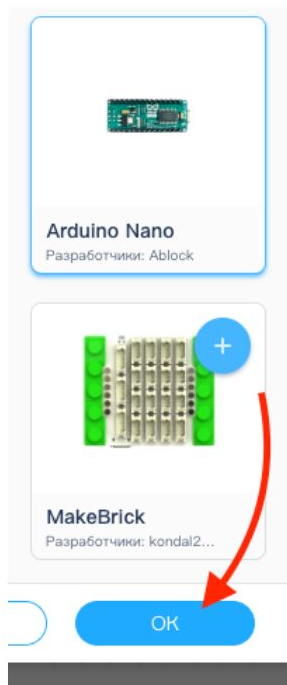
Удалим Codey, добавим Arduino Nano



## Шаг 2

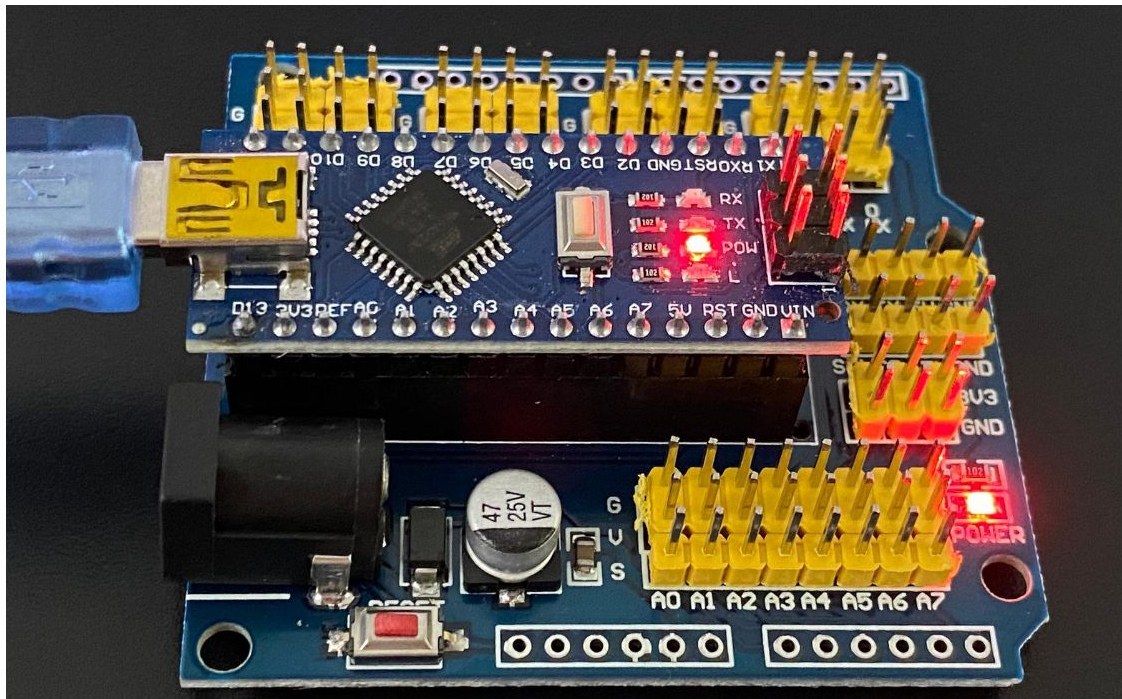
# Шаг 3

Выбрав плату с которой мы будем работать, нажимаем ОК. Она должна появиться во вкладке устройства





Установим Arduino Nano к плате расширения, как показано на рисунке, подключив её кабелем USB-mini из комплекта к компьютеру



Шаг 4

# Шаг 5

Подключите свое устройство

[Как использовать устройство?](#)

Переключатель режимов ?

Перезагрузить в Жить

Подключение

Соединим плату с компьютером  
Название порта может отличаться

USB

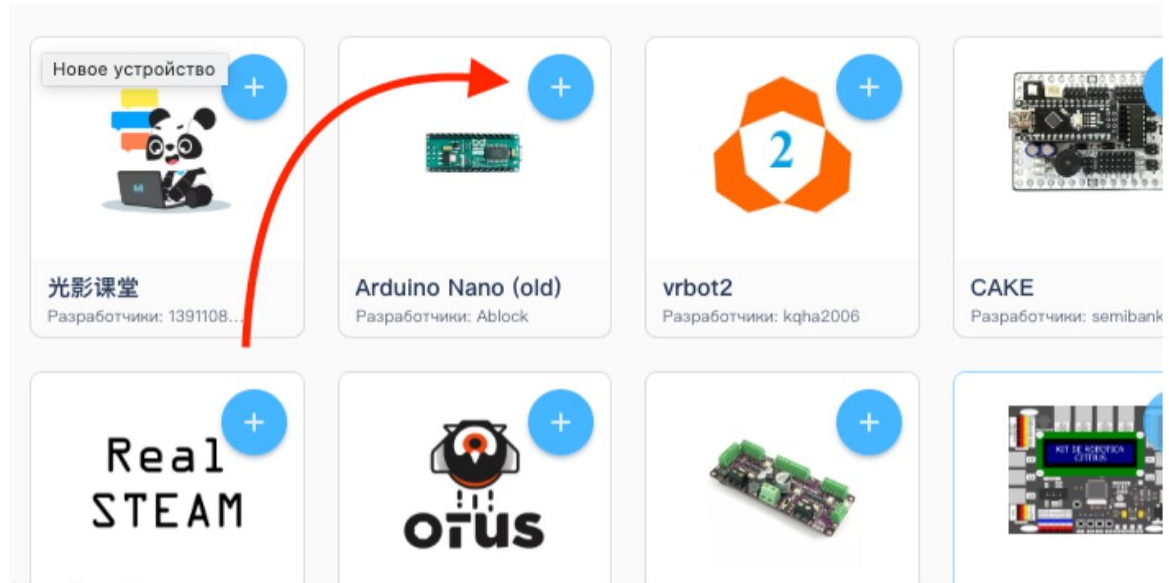
Показать все подключенные устройства

/dev/tty.usbserial-1420

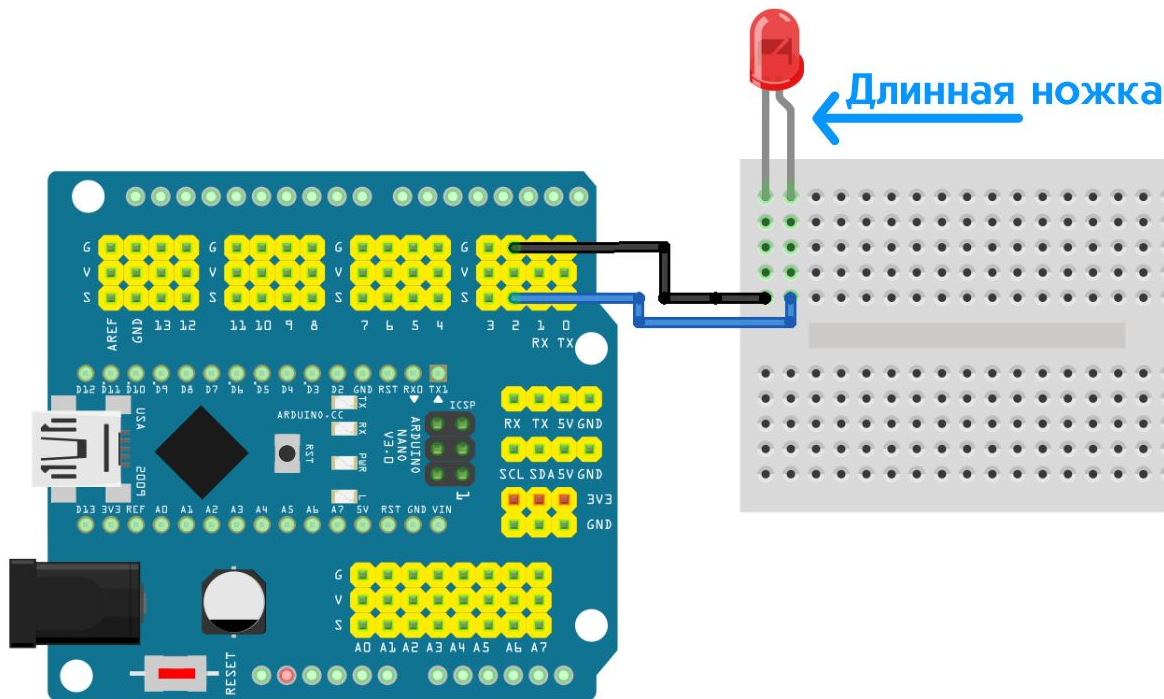
**Подключение**

- Убедитесь, что кабель USB правильно подключен к устройству.
- Убедитесь, что подключенное устройство включено.
- Только одно устройство может быть подключено одновременно в этой версии. Таким образом, подключение этого устройства приведет к разъединению предыдущего.

Может быть такое, что ваша Arduino Nano не подключается к компьютеру, тогда следует выбрать в качестве используемого устройства с другой версией загрузчика Arduino nano (Old)



Соберем схему с применением макетной платы, пары соединительных проводов, светодиодом и Arduino Nano. Короткую ногу светодиода соединим с GND платы, а длинную с 2 портом



# Шаг 6

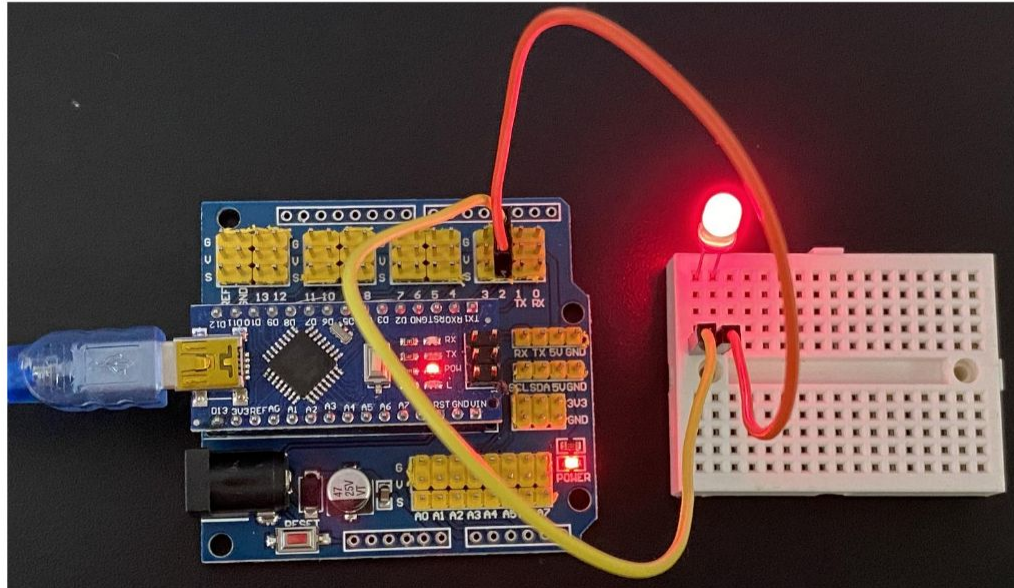
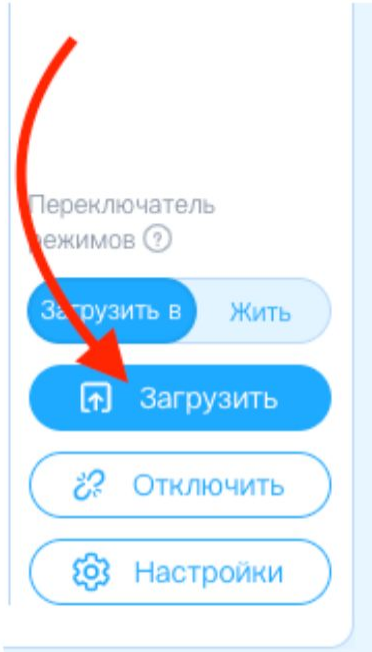
# Шаг 7

Напишем код для мигания светодиодом, с частотой 1 секунда.



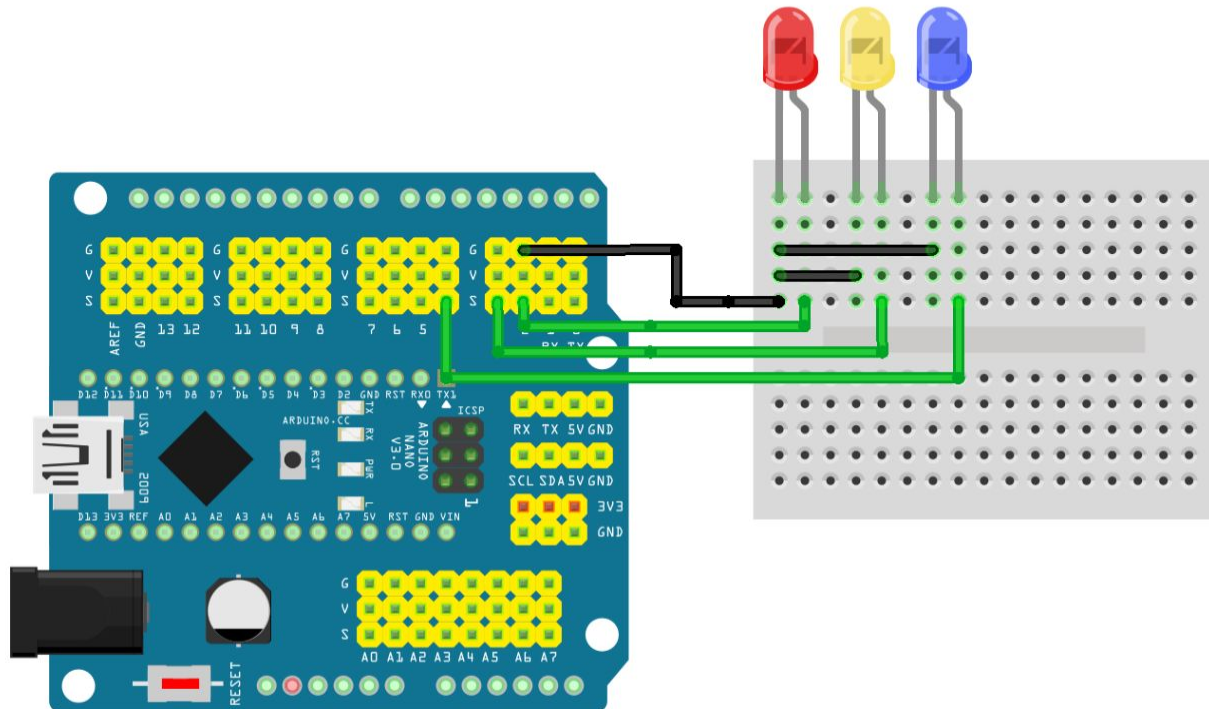
Что бы Arduino выполняла написанный нами блокный код, нужно загрузить его. Поставьте вместо 1 значения 0,5, 2, 5 и последите как измениться частота мигания

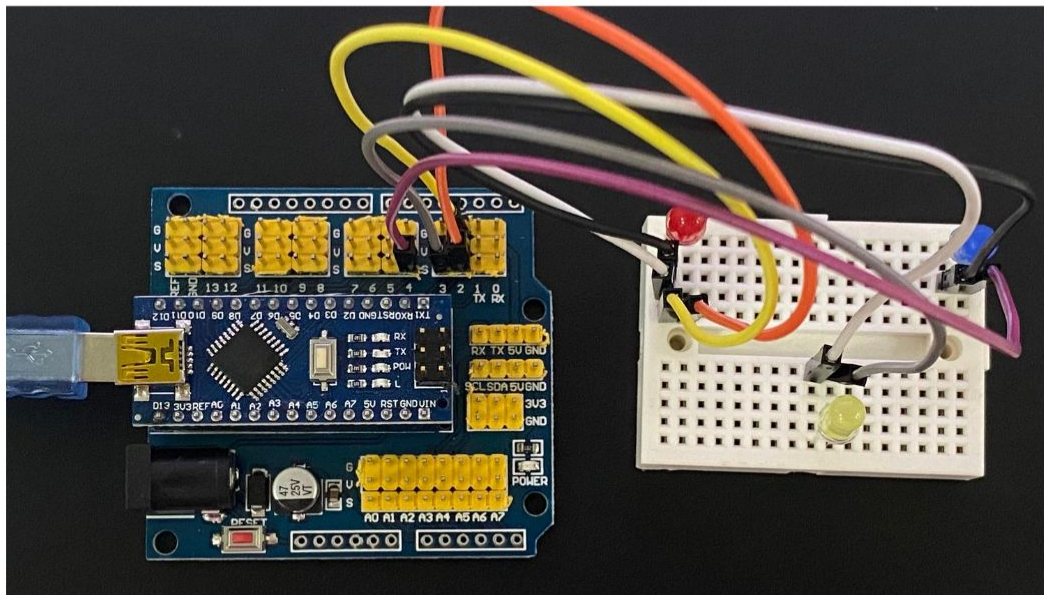
# Шаг 8



# Шаг 9

Соберем схему с тремя светодиодами





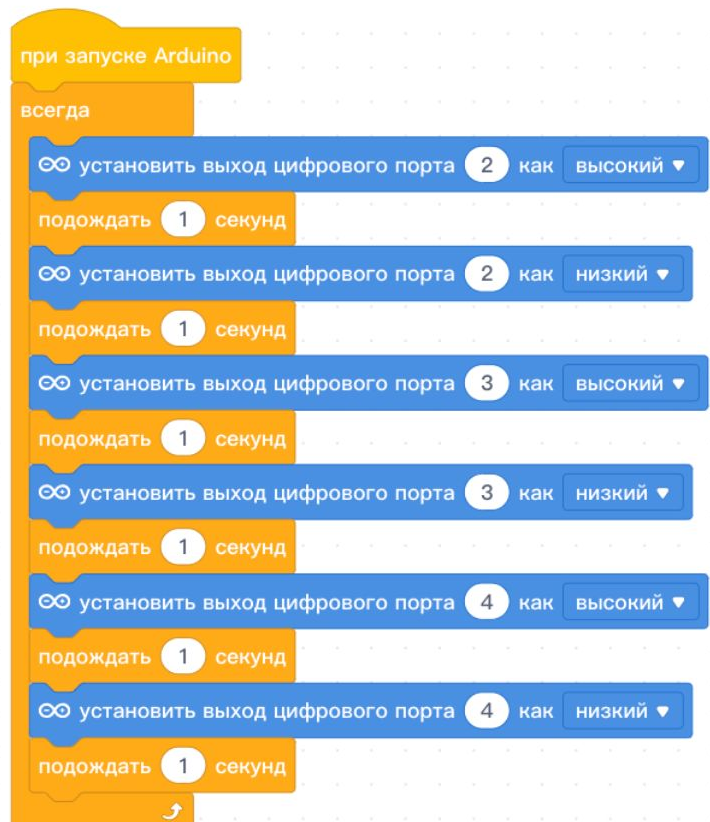
Конечно, расположение элементов на макетной плате не играет никакой роли. Самое главное - сохранять правильность подключений по схеме.

Цвета соединительных проводов никак не влияют на работу электрических схем. Для большего удобства в электронике принято правило обозначать земляные провода (G) черным цветом, сигнальные (S, A0) **желтым** или **синим**, а провода плюсового питания (+5V) **красным**.

Землю можно взять из отдельных металлических контактов - пинов, обозначенных (G). В нашем случае мы пользуемся так называемой "общей" землей



Напишем два программных кода, один из них будет поочередно зажигать и выключать светодиоды, а второй - давать команду светодиодам работать синхронно. Важно не забывать загружать внесенные в код изменения в плату



```
при запуске Arduino
всегда
  установить выход цифрового порта 2 как высокий
  подождать 1 секунд
  установить выход цифрового порта 2 как низкий
  подождать 1 секунд
  установить выход цифрового порта 3 как высокий
  подождать 1 секунд
  установить выход цифрового порта 3 как низкий
  подождать 1 секунд
  установить выход цифрового порта 4 как высокий
  подождать 1 секунд
  установить выход цифрового порта 4 как низкий
  подождать 1 секунд
```

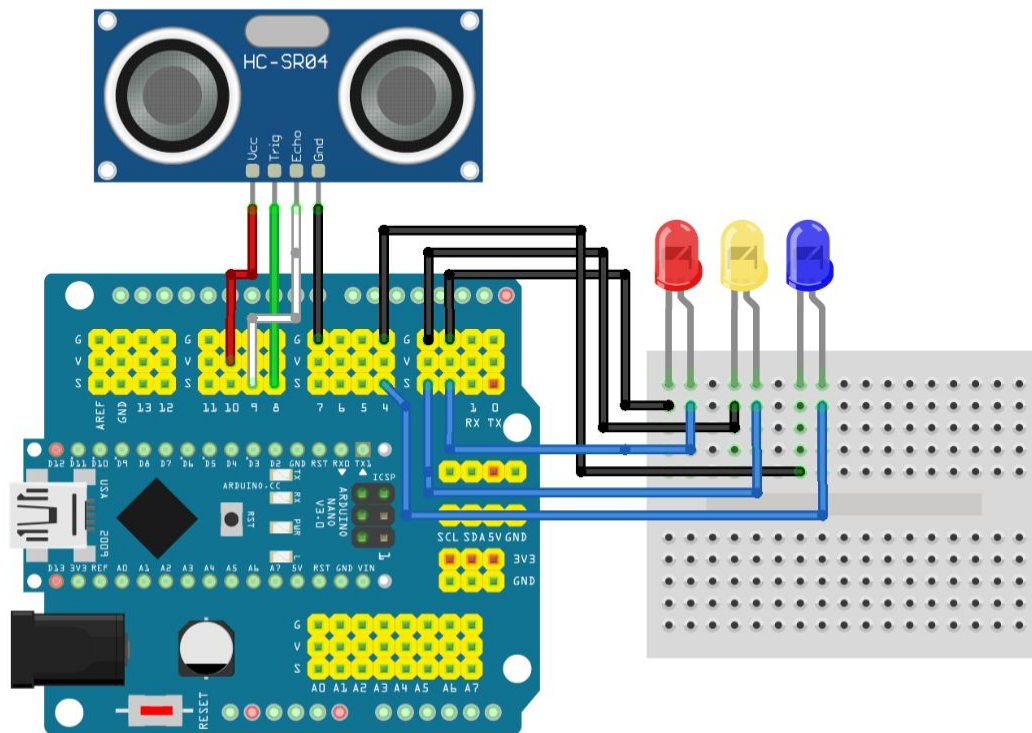
# Шаг 10

# Шаг 11

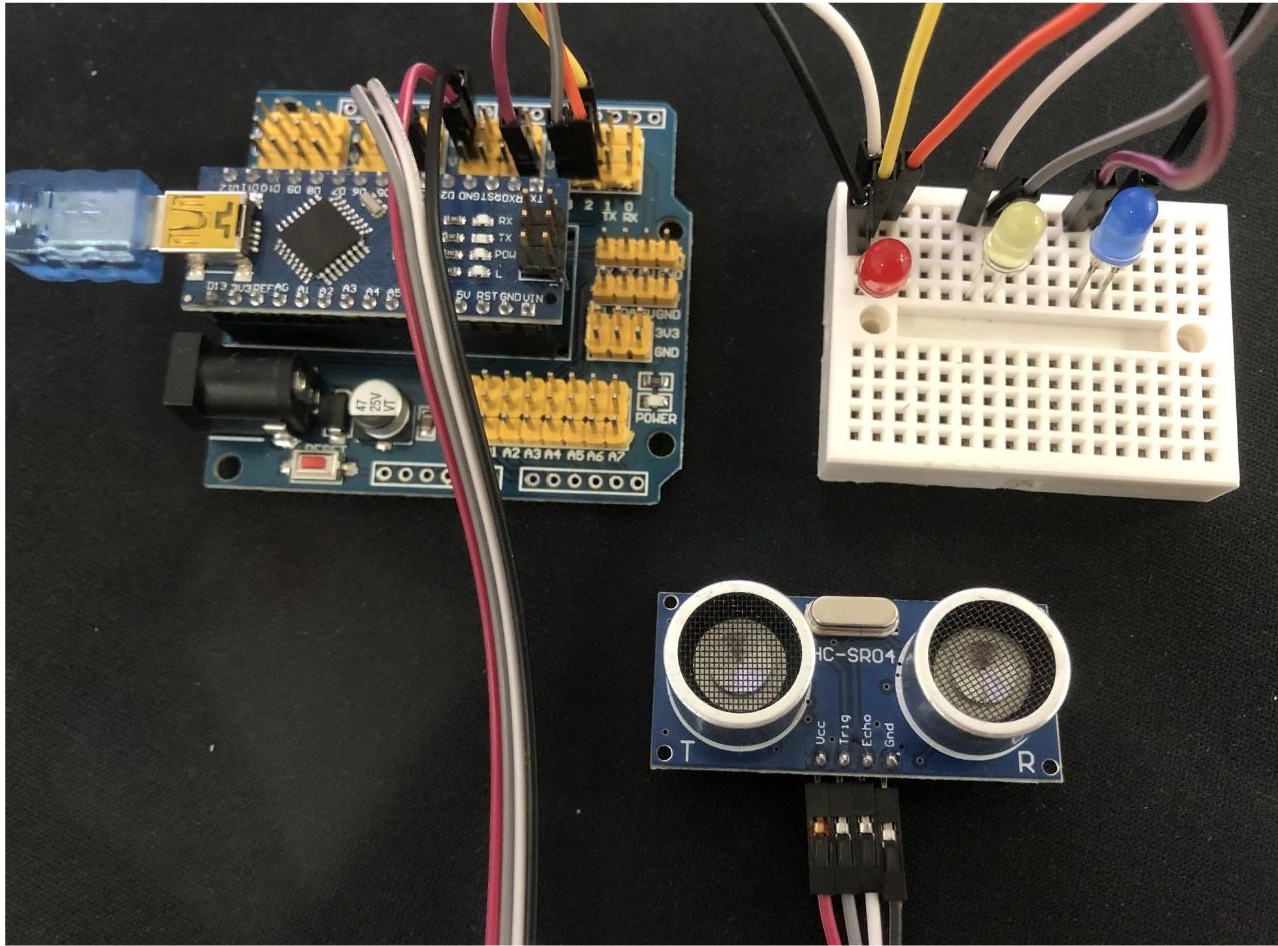
Разная периодичность работы зависит от логического алгоритма, с которым расставлены задержки(подождать)

```
при запуске Arduino
всегда
  ∞ установить выход цифрового порта 2 как высокий
  ∞ установить выход цифрового порта 3 как высокий
  ∞ установить выход цифрового порта 4 как высокий
  подождать 1 секунд
  ∞ установить выход цифрового порта 2 как низкий
  ∞ установить выход цифрового порта 3 как низкий
  ∞ установить выход цифрового порта 4 как низкий
  подождать 1 секунд
```

Следующая схема будет включать светодиоды, в зависимости от дальности предмета перед глазами нашего робота - ультразвукового датчика расстояния HC-SR04



# Шаг 12



Считывать данные мы будем с порта Trig который подключим к 8 пину и порта Echo который подключим к 9 пину. Нам понадобятся новые блоки из набора блоков "Датчик" и "Оператор"

Категории в палитре:

- Контакт
- последовательный порт
- Данные
- Датчик**
- События
- Управление

Блоки в рабочем пространстве:

- считать данные с портов trig 1 echo 1 ультразвукового датчика
- таймер
- обнулить таймер
- при запуске Arduino
- всегда

Категории в палитре:

- Контакт
- последовательный порт
- Данные
- Датчик
- События
- Управление
- Оператор

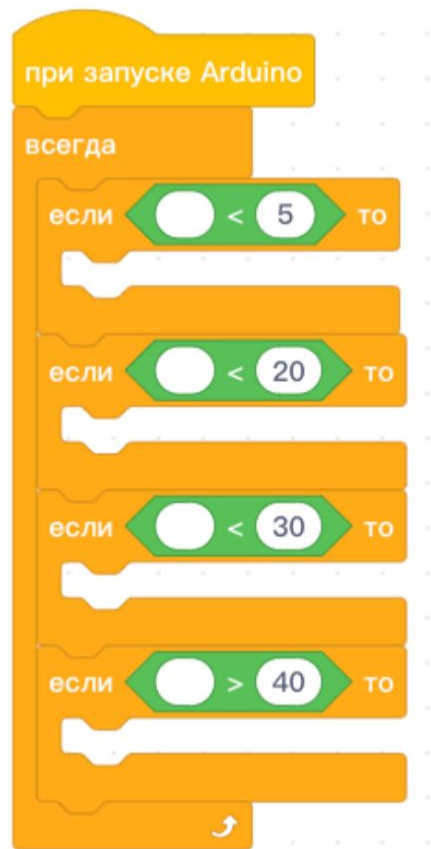
Блоки в категории Датчик:

- > 50
- < 50
- = 50

Блоки в категории Оператор:

- и
- или
- не

# Шаг 13



Логика работы алгоритма очень простая - если значение с датчика меньше указанного значения, то включаются или выключаются нужные нам светодиоды

Добавим процесс считывания показаний датчика расстояния



```
при запуске Arduino
всегда
если << считывать данные с портов trig 8 echo 9 ультразвукового датчика < 5 >> то
  [ ]
если << считывать данные с портов trig 8 echo 9 ультразвукового датчика < 20 >> то
  [ ]
если << считывать данные с портов trig 8 echo 9 ультразвукового датчика < 30 >> то
  [ ]
если << считывать данные с портов trig 8 echo 9 ультразвукового датчика > 40 >> то
  [ ]
```

The image shows a Scratch script for an Arduino project. It starts with a 'when Arduino starts' block, followed by a 'forever' loop. Inside the loop, there are four 'if' blocks, each with a condition to read data from a distance sensor (trig pin 8, echo pin 9). The conditions are: if distance is less than 5, less than 20, less than 30, and greater than 40. Each 'if' block has a corresponding empty block below it, likely for displaying the distance.



# Шаг 14

# Шаг 15

```
при запуске Arduino
всегда
если [считать данные с портов trig 8 echo 9 ультразвукового датчика < 5] то
  [установить выход цифрового порта 4 как высокий]
  [установить выход цифрового порта 3 как высокий]
  [установить выход цифрового порта 2 как высокий]
если [считать данные с портов trig 8 echo 9 ультразвукового датчика < 20] то
  [ ]
если [считать данные с портов trig 8 echo 9 ультразвукового датчика < 30] то
  [ ]
если [считать данные с портов trig 8 echo 9 ультразвукового датчика > 40] то
  [ ]

```



при запуске Arduino

всегда

если считывать данные с портов trig 8 echo 9 ультразвукового датчика < 5 то

установить выход цифрового порта 4 как высокий

установить выход цифрового порта 3 как высокий

установить выход цифрового порта 2 как высокий

если считывать данные с портов trig 8 echo 9 ультразвукового датчика < 20 то

установить выход цифрового порта 4 как низкий

установить выход цифрового порта 3 как высокий

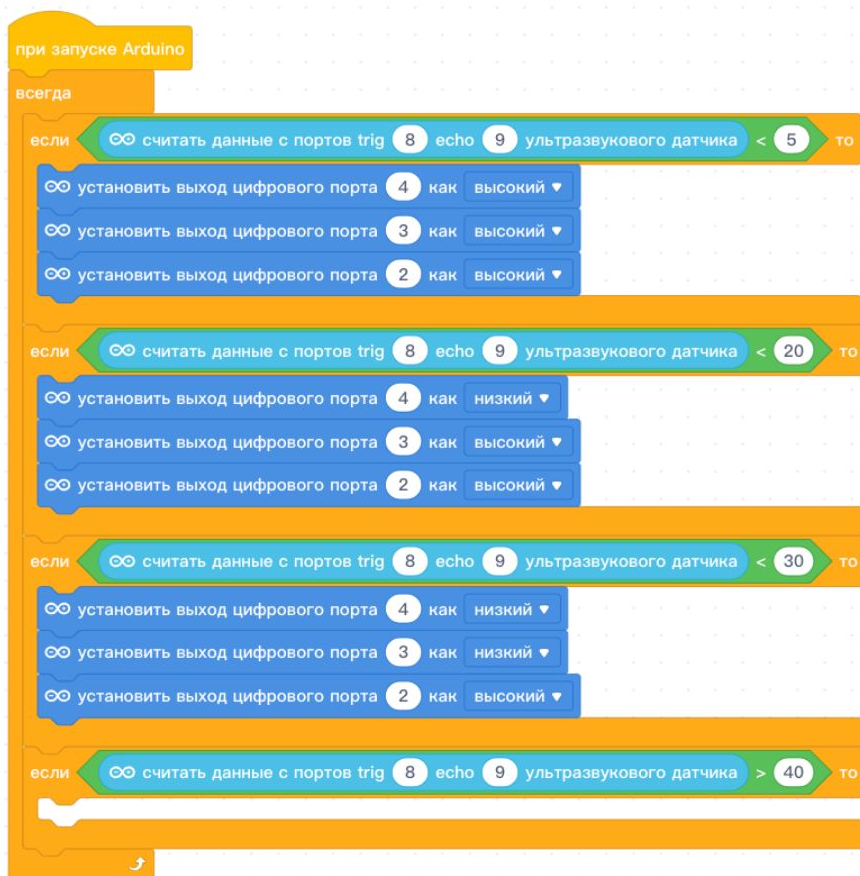
установить выход цифрового порта 2 как высокий

если считывать данные с портов trig 8 echo 9 ультразвукового датчика < 30 то

если считывать данные с портов trig 8 echo 9 ультразвукового датчика > 40 то

# Шаг 16

# Шаг 17



при запуске Arduino

всегда

если  считать данные с портов trig 8 echo 9 ультразвукового датчика < 5 то

установить выход цифрового порта 4 как высокий ▾

установить выход цифрового порта 3 как высокий ▾

установить выход цифрового порта 2 как высокий ▾

если  считать данные с портов trig 8 echo 9 ультразвукового датчика < 20 то

установить выход цифрового порта 4 как низкий ▾

установить выход цифрового порта 3 как высокий ▾

установить выход цифрового порта 2 как высокий ▾

если  считать данные с портов trig 8 echo 9 ультразвукового датчика < 30 то

установить выход цифрового порта 4 как низкий ▾

установить выход цифрового порта 3 как низкий ▾

установить выход цифрового порта 2 как высокий ▾

если  считать данные с портов trig 8 echo 9 ультразвукового датчика > 40 то

установить выход цифрового порта 4 как низкий ▾

установить выход цифрового порта 3 как низкий ▾

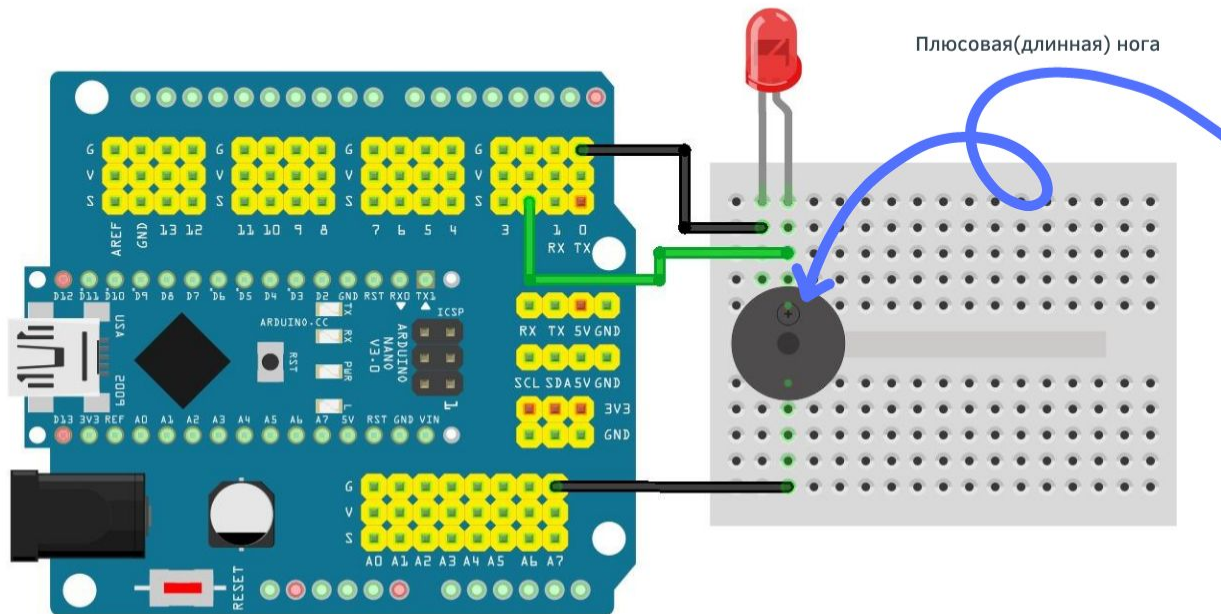
установить выход цифрового порта 2 как низкий ▾

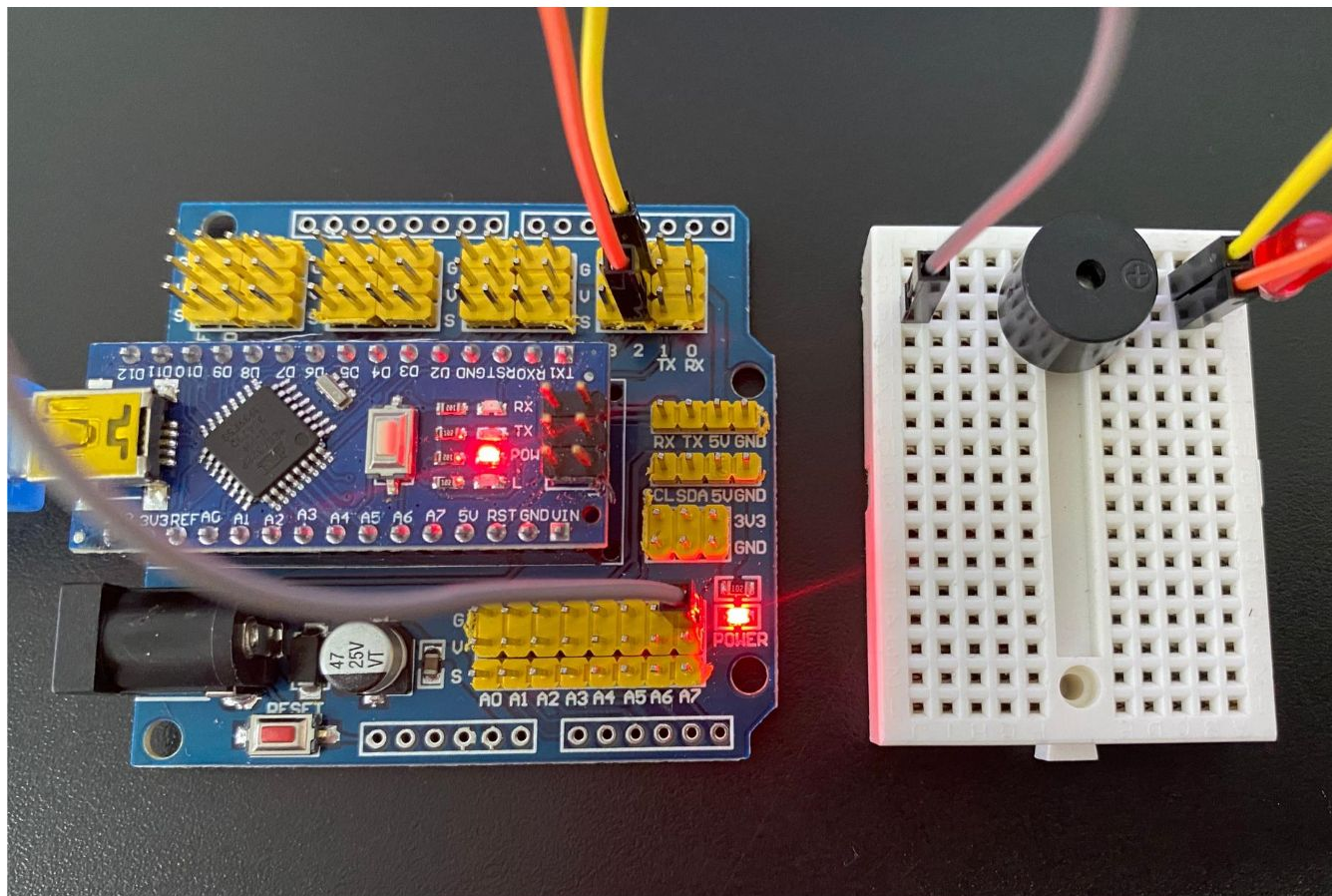
# Шаг 18

# Шаг 19

Соберем схему с пьезопищалкой и светодиодом, которая позволит проигрывать ноты различного звучания и частоты.

Светодиод будет выступать в качестве индикации наличия сигнала





В программном коде можно попробовать составить мелодию из различных нот и разной длительности.



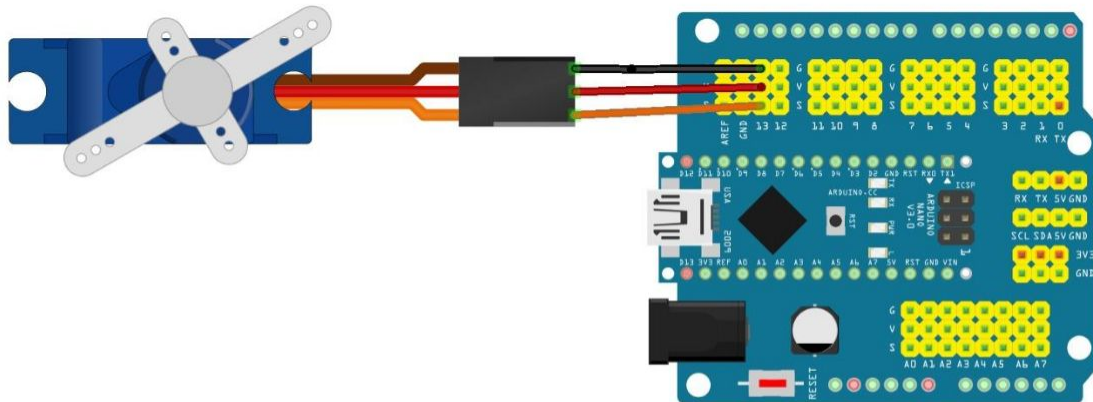
```
при запуске Arduino
повторить 1
  порт 2 проигрывает ноту C4 0.25 долей
  подождать 0.25 секунд
  порт 2 проигрывает ноту E4 0.5 долей
  порт 2 проигрывает ноту C4 0.25 долей
  подождать 0.25 секунд
  порт 2 проигрывает ноту E4 0.5 долей
  порт 2 проигрывает ноту G4 0.25 долей
  подождать 0.25 секунд
  порт 2 проигрывает ноту C4 1 долей
  подождать 0.25 секунд
```

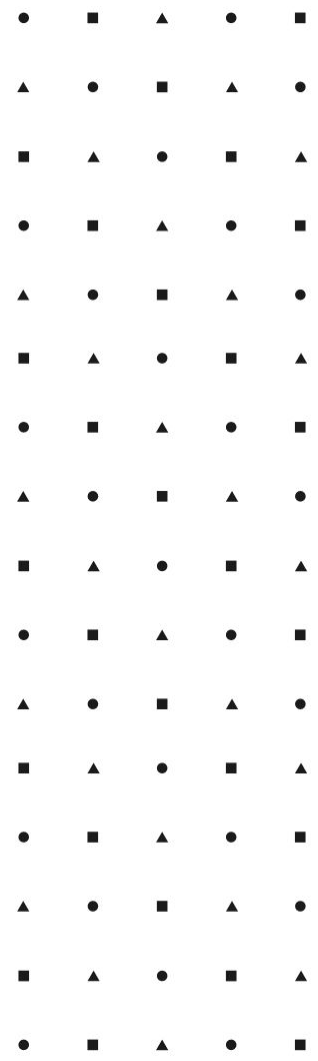
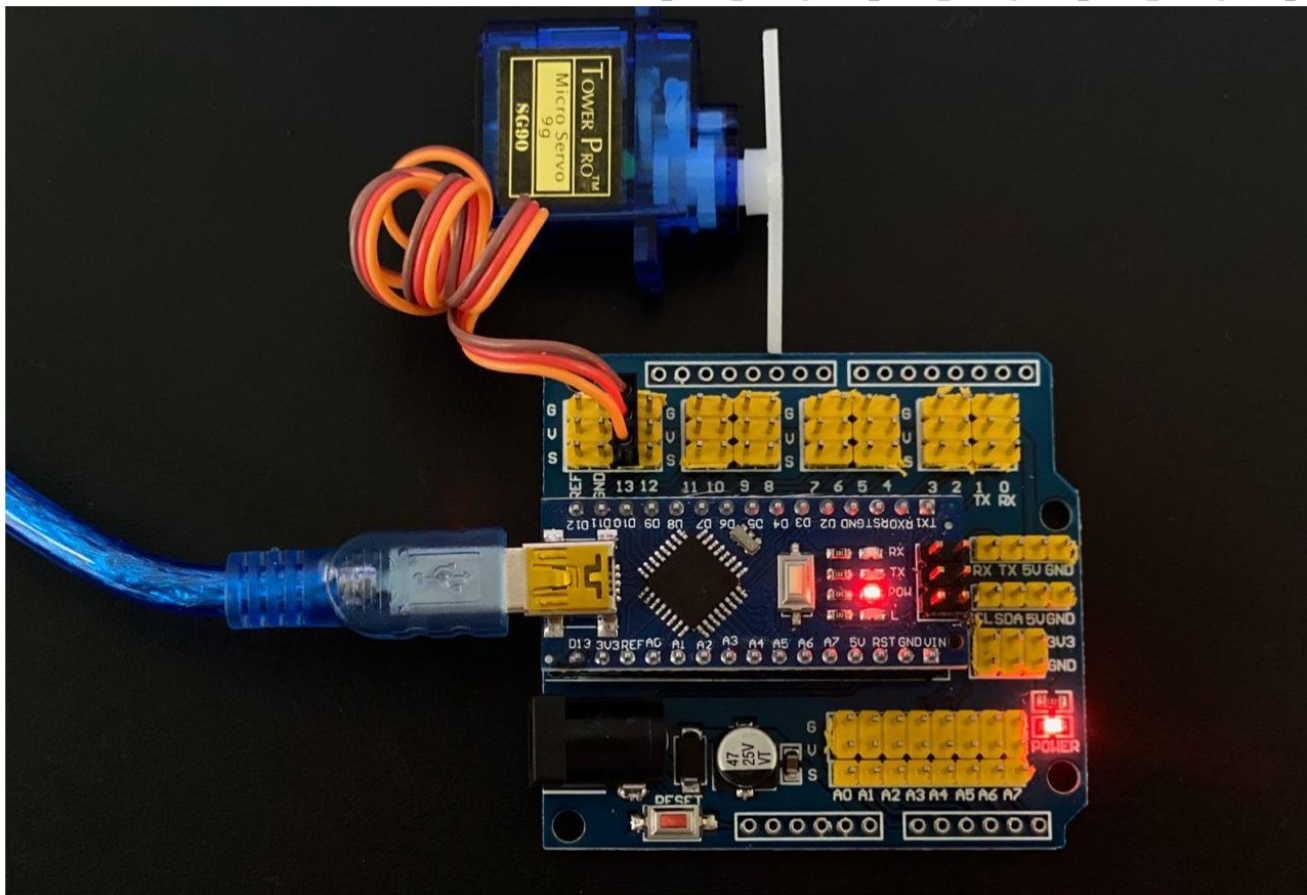
# Шаг 20

# Шаг 21

Схема подключения сервопривода очень простая. Но есть 2 варианта подключения: с помощью соединительных проводов или прямого подключения шлейфа сервопривода к плате расширения. При сборке робота сервопривод будет подключаться напрямую, без соединительных проводов.

Важно не перепутать полярность подключения - оранжевый провод подключается в S, красный в V, коричневый (черный) – в G.

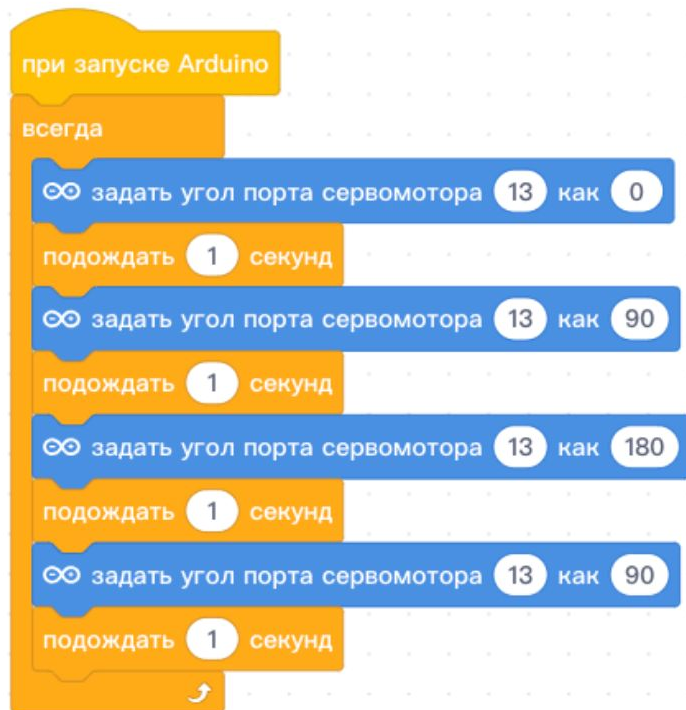






Программный код будет поворачивать вал сервопривода в 3 ключевых положения: 180, 90, 0 градусов.

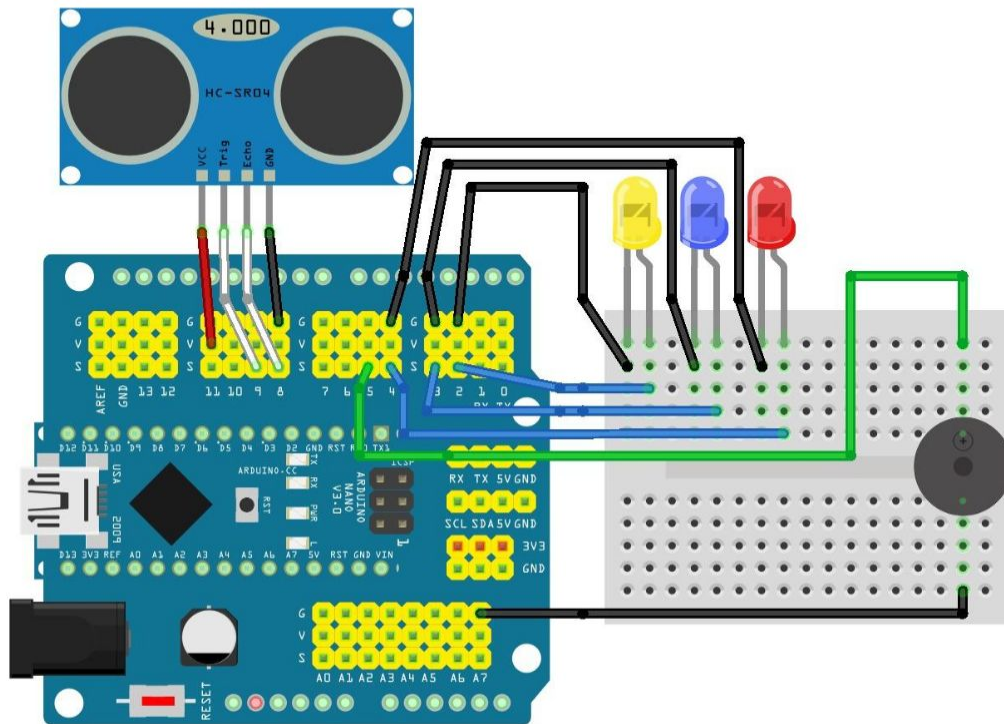
Обязательно добавляем задержку между углами поворота: без неё сервопривод физически не будет успевать менять положение

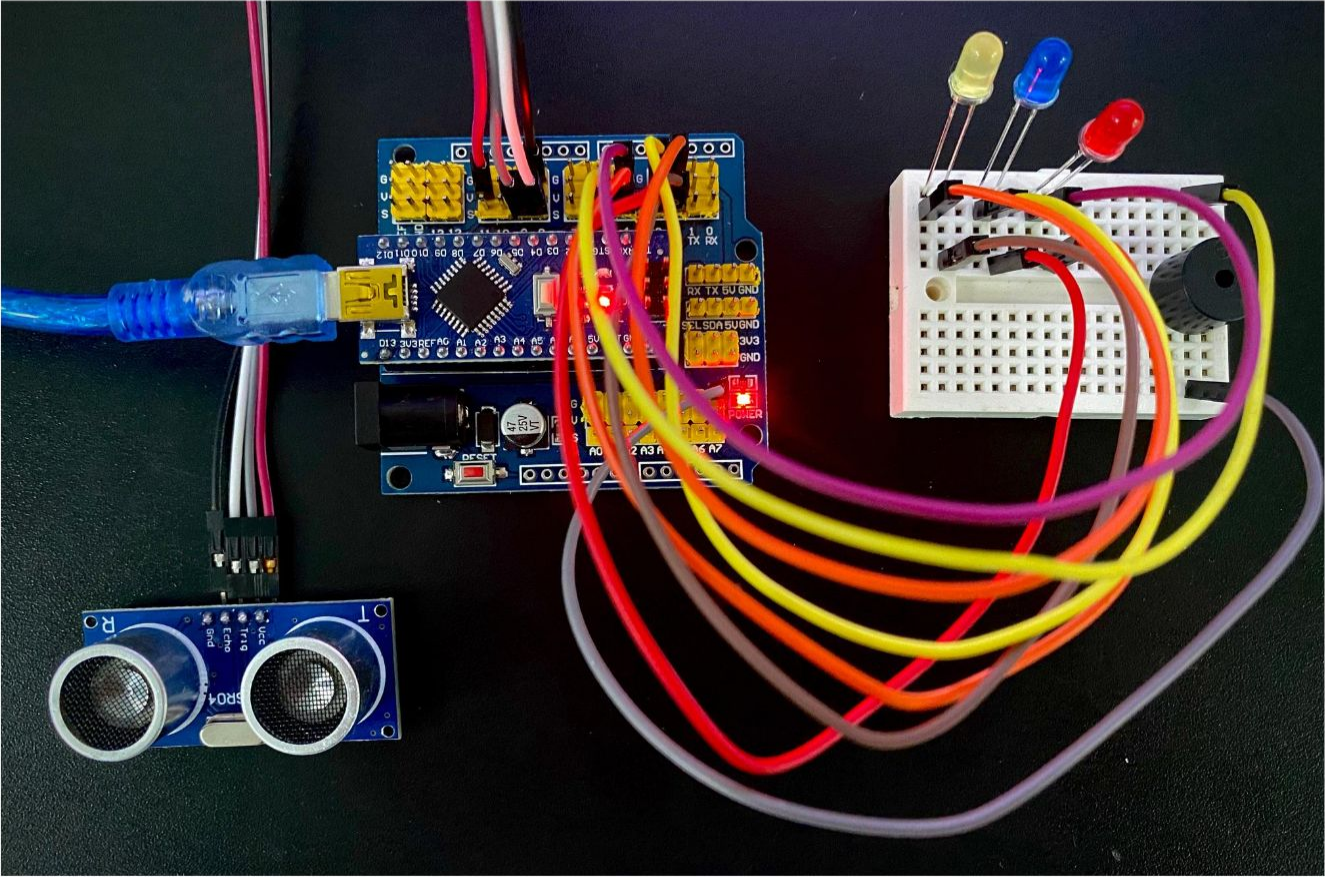


## Шаг 22

# Шаг 23

Финальной схемой перед сборкой Отто робота будет схема настоящего Парктроника - устройства которое считывает расстояние от заднего бампера машины до бордюра и подает световые и звуковые сигналы





Напишем блокный программный код, реализующий нашу задумку

```
при запуске Arduino
всегда
  если <math>\infty</math> считать данные с портов trig 9 echo 8 ультразвукового датчика < 30 то
  иначе
  если <math>\infty</math> считать данные с портов trig 9 echo 8 ультразвукового датчика < 15 то
  иначе
  если <math>\infty</math> считать данные с портов trig 9 echo 8 ультразвукового датчика < 4 то
  иначе
  ждать 0.3 секунд
```

# Шаг 24

# Шаг 25

```
при запуске Arduino
всегда
если < считывать данные с портов trig 9 echo 8 ультразвукового датчика < 30 > то
  установить выход цифрового порта 2 как высокий
иначе
  установить выход цифрового порта 2 как низкий
если < считывать данные с портов trig 9 echo 8 ультразвукового датчика < 15 > то
  установить выход цифрового порта 3 как высокий
иначе
  установить выход цифрового порта 3 как низкий
если < считывать данные с портов trig 9 echo 8 ультразвукового датчика < 4 > то
  установить выход цифрового порта 4 как высокий
  порт 5 проигрывает ноту C4 0.25 долей
иначе
  установить выход цифрового порта 4 как низкий
подождать 0.3 секунд
```



**Поздравляем!  
Можно смело  
переходить к  
сборке Отто  
робота**

