

# Пульсар

## Эксперимент №6

В этом эксперименте мы плавно наращиваем яркость светодиодной шкалы, управляя большой нагрузкой через транзистор.

Педагог дополнительного образования

МБУДО «РЦВР» Юрышев Д.Э.

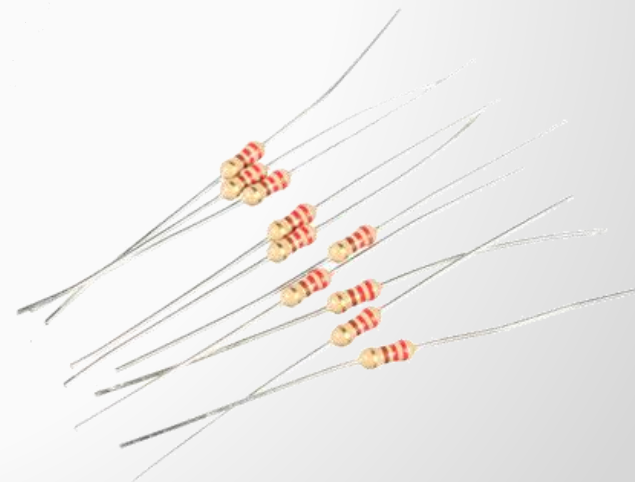
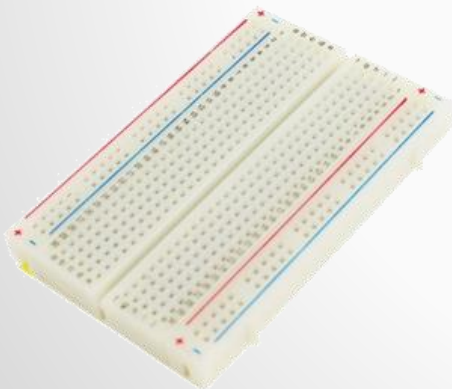
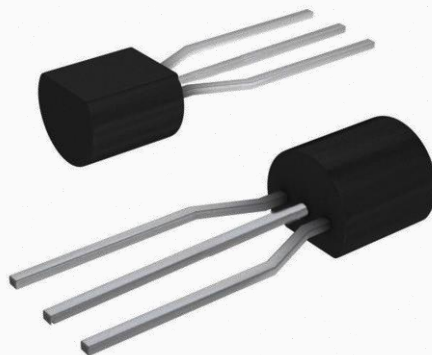
# Прочтите перед выполнением

- Биполярный транзистор
- Светодиодные сборки

# Детали для эксперимента



www.mikrobit.ru

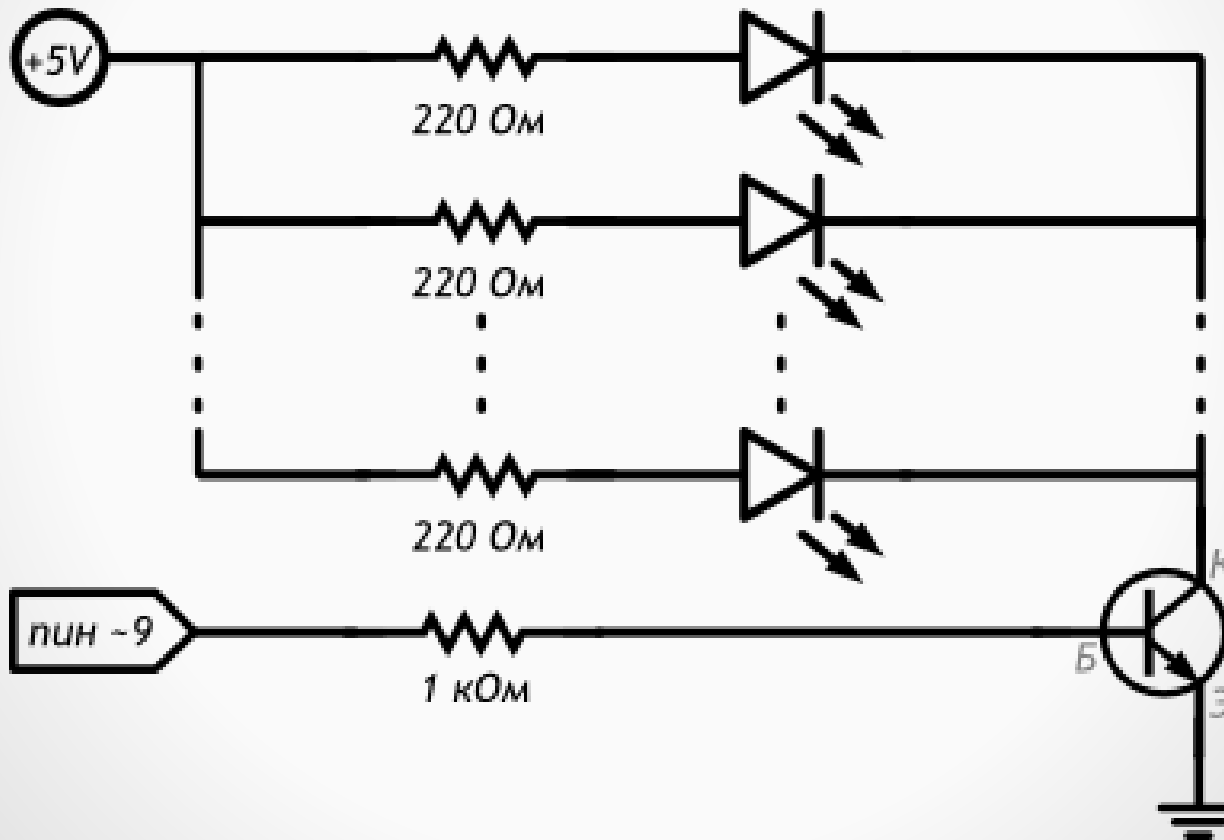


20.07.2019

Эксперимент №6 «Пульсар»

3

# Принципиальная схема

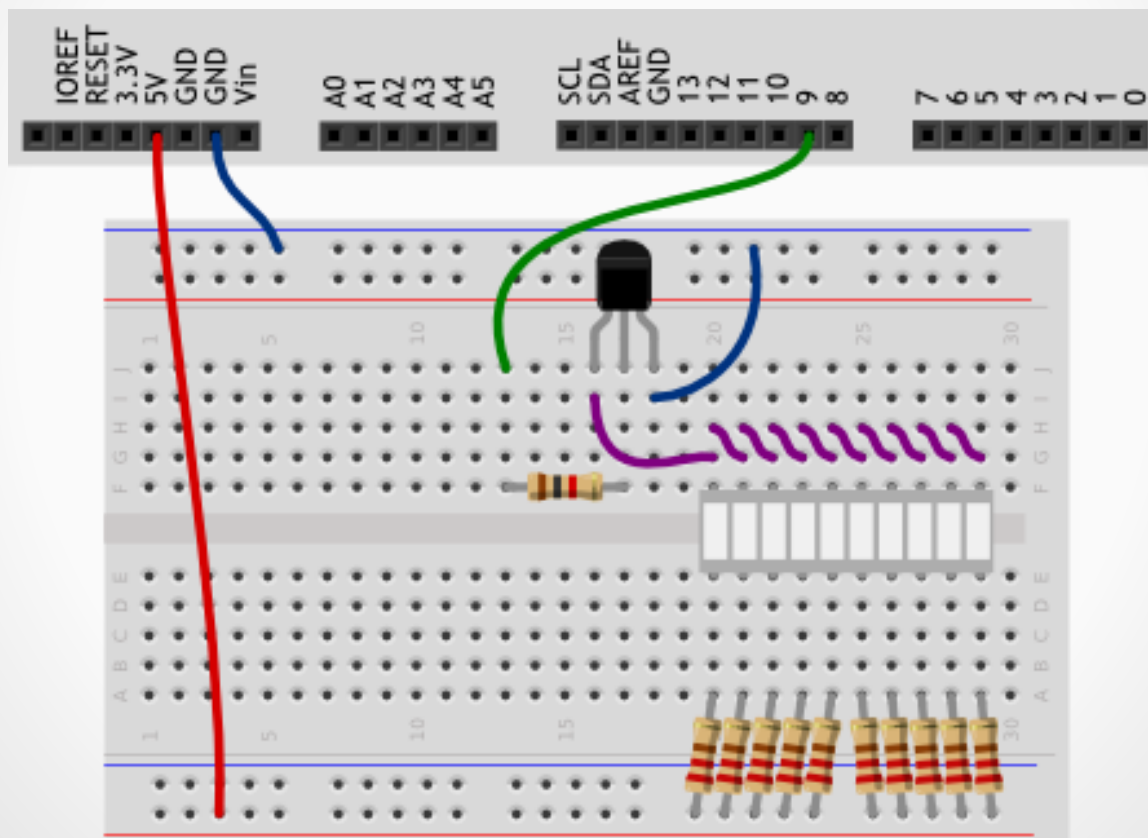


20.07.2019

Эксперимент №6 «Пульсар»

4

# Схема на макетке





# Обратите внимание

- Светодиодная шкала — это несколько светодиодов в одном корпусе. Нам нужно чтобы питание шло к их анодам, а катоды направлялись к земле. Скорее всего на вашей шкале аноды находятся со стороны маркировки. Если шкала не светится, когда должна, попробуйте перевернуть ее.
- База биполярного транзистора — это его средняя ножка. Если повернуть транзистор плоской стороной к себе, ножками вниз, то левая ножка это коллектор, а правая — эмиттер.
- Если эту схему собрать без резистора между базой транзистора и портом Arduino, мы практически устроим короткое замыкание порта на землю. Рано или поздно это выведет из строя транзистор или ножку микроконтроллера.
- Зачем здесь вообще транзистор? Без него такое количество светодиодов будет потреблять больше тока, чем 40 мА, которые может себе позволить цифровой пин платы. Поэтому мы берем питание из порта 5V, рассчитанного на ток до 500 мА, а на цифровой порт ставим транзистор, чтобы с помощью малого тока управлять большим.
- В данном случае мы включили 10 светодиодов параллельно, каждый через отдельный резистор. Включать их через один резистор неправильно: даже светодиоды из одной партии имеют минимальный разброс вольт-амперных характеристик, вследствие чего они:
  - Светились бы с различной яркостью
  - Из-за минимальной разницы во времени включения, больший ток, прошедший через первый включившийся светодиод, мог бы вывести его из строя. И так по цепочке.

# Скетч

```
p060_pulse_bar.ino
```

```
#define CONTROL_PIN 9
```

```
// переменные верхнего уровня, т.е. объявленные вне функций, называют глобальными. Их значения сохраняются всё время, пока работает микроконтроллер  
int brightness = 0;
```

```
void setup()
```

```
{  
  pinMode(CONTROL_PIN, OUTPUT);  
}
```

```
void loop()
```

```
{  
  // увеличиваем значение яркости на единицу, чтобы нарастить яркость. Однако яркость не должна быть более 255, поэтому используем операцию остатка от деления, чтобы при достижении значения 255, следующим значением снова стал 0  $Y \% X$  – это остаток от деления  $Y$  на  $X$ ; плюс, минус, делить, умножить, скобки – как в алгебре.
```

```
  brightness = (brightness + 1) % 256;
```

```
  // подаём вычисленный ШИМ-сигнал яркости на пин с базой управляющего транзистора
```

```
  analogWrite(CONTROL_PIN, brightness);
```

```
  // ждём 10 мс перед следующим наращиванием яркости. Таким образом, полный накал будет происходить в течение  $256 \times 10 = 2560$ 
```

```
мс  
  delay(10);  
}
```

# Пояснения к коду

Как мы уже знаем, `analogWrite(pin, value)` в качестве `value` принимает значения от 0 до 255. Если передать значение из-за пределов этого диапазона, функция сработает, но в общем случае вы получите неожиданный результат.

Оператор `X % Y` дает остаток от деления `X` на `Y`. Если `X` меньше `Y`, т.е. целая часть результата деления равна 0, оператор `%` будет возвращать `X`. Таким образом:

Пока `brightness + 1` меньше 256, в `brightness` записывается значение `brightness + 1`

Как только `brightness + 1` принимает значение 256, результатом `(brightness + 1) % 256` становится 0 и на следующей итерации `loop()` всё начинается сначала.

Оператор `%` работает только с целыми операндами.

В выражении `(brightness + 1) % 256` скобки используются для назначения порядка действий. Операция `%` имеет больший приоритет, чем `+`, а сложение нам нужно выполнять раньше. С операциями умножения и деления оператор взятия остатка имеет одинаковый приоритет.




# Вопросы для проверки себя

1. Почему у светодиодной шкалы на 10 сегментов 20 ножек?
2. Зачем в схеме биполярный транзистор?
3. За счет чего увеличивается яркость шкалы?
4. Почему после достижения значения 255 переменная `brightness` обнуляется?

# Задания для самостоятельного решения

1. Измените программу так, чтобы яркость шкалы росла только до половины от максимальной.
2. Измените программу так, чтобы шкала становилась максимально яркой в три раза быстрее, без изменения функции `delay`.
3. Измените исходную программу так, чтобы такой же результат был получен без использования операции `%`, но с применением условного оператора `if`.



**Единственное условие, от  
которого зависит успех,  
есть терпение.**

**Толстой Лев Николаевич**