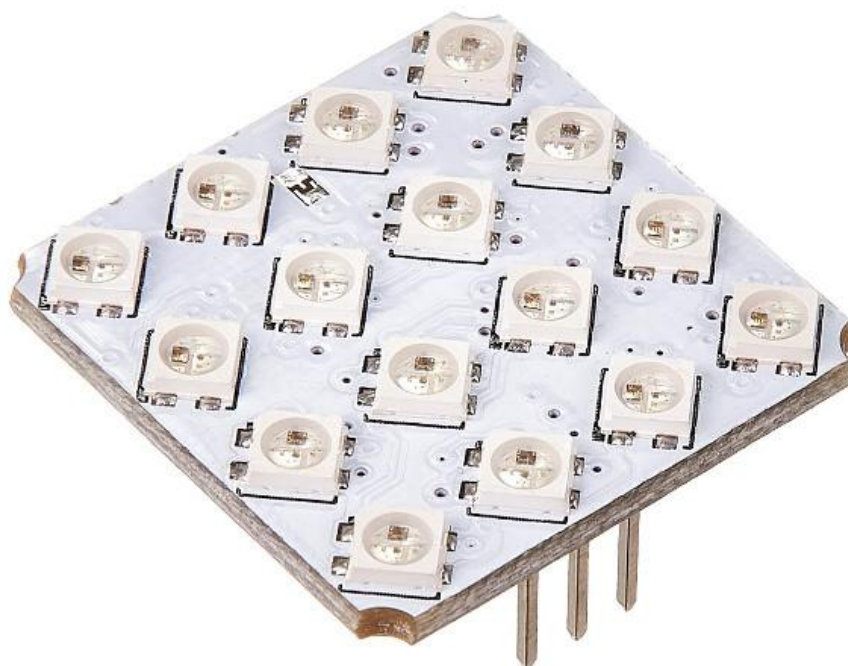


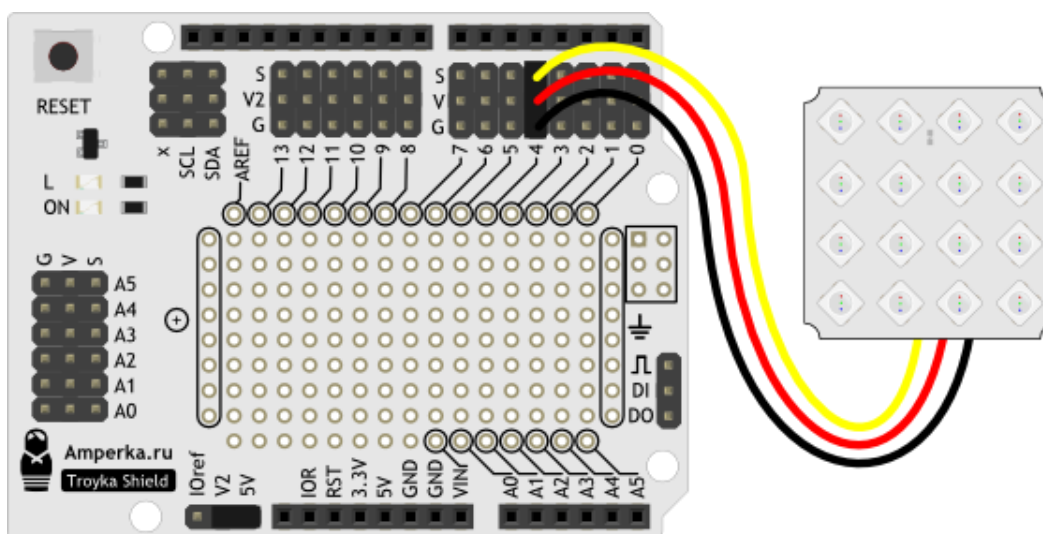
# Светодиодная RGB матрица 4×4 (Тройка-модуль)



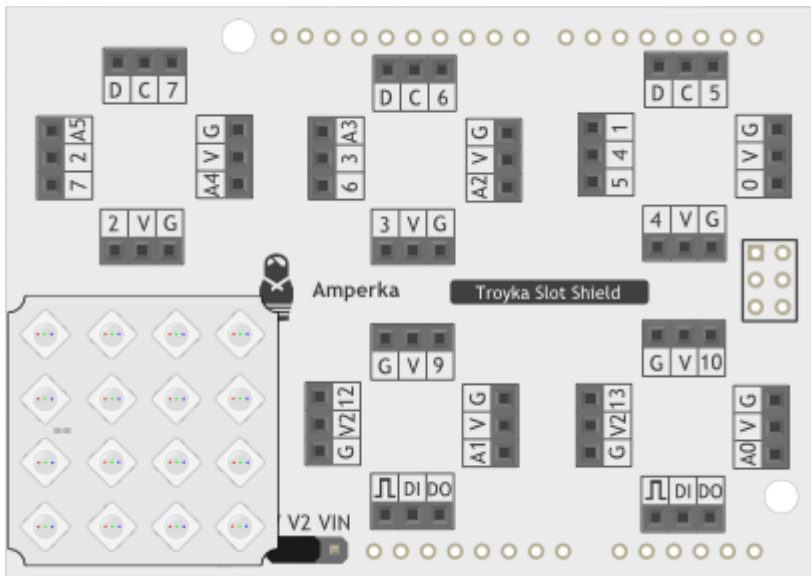
[RGB Matrix](#) — цепочка из шестнадцати цветных светодиодов, собранная на одноюнитовом модуле.

## Подключение и настройка

RGB-матрица общается с управляющей электроникой по [трём проводам](#). При подключении к [Arduino](#) или [Iskra JS](#) удобно использовать [Troyka Shield](#).



С [Troyka Slot Shield](#) можно обойтись без лишних проводов.



## Примеры работы

### Пример работы для Arduino

[matrixRGB.ino](#)

```
// библиотека для работы с RGB-матрицей
#include <Adafruit_NeoPixel.h>

// номер пина, к которому подключена RGB-матрица
#define MATRIX_PIN 4
// количество светодиодов в матрице
#define LED_COUNT 16

// создаём объект класса Adafruit_NeoPixel
Adafruit_NeoPixel matrix = Adafruit_NeoPixel(LED_COUNT, MATRIX_PIN,
NEO_GRB + NEO_KHZ800);

void setup()
{
  // инициализация RGB-матрицы
  matrix.begin();
}

void loop()
{
  // заполняем RGB-матрицу по сегментам «бегущий огонь» красного цвета
  colorWipe(matrix.Color(255, 0, 0), 50);
  // заполняем RGB-матрицу по сегментам «бегущий огонь» зелёного цвета
  colorWipe(matrix.Color(0, 255, 0), 50);
  // заполняем RGB-матрицу по сегментам «бегущий огонь» синего цвета
  colorWipe(matrix.Color(0, 0, 255), 50);
  // гасим RGB-матрицу по сегментам «бегущая тень»
  colorWipe(matrix.Color(0, 0, 0), 50);
}

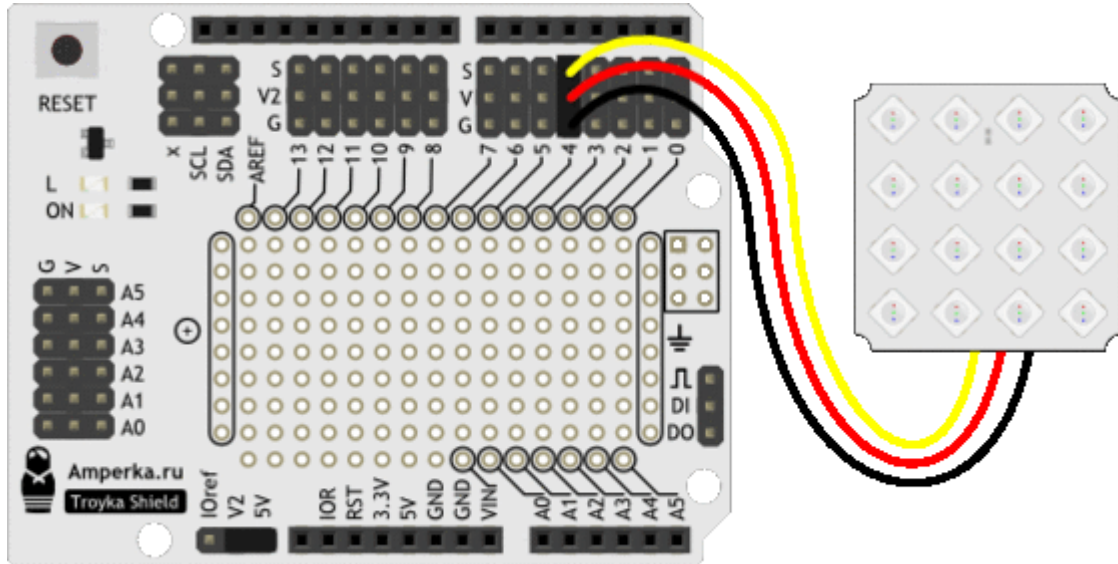
// функция заполнения каждого сегмента
void colorWipe(uint32_t c, uint8_t wait)
{
  for (uint16_t i = 0; i < matrix.numPixels(); i++) {
    // заполняем текущий сегмент выбранным цветом
    matrix.setPixelColor(i, c);
    matrix.show();
  }
}
```

```

// ждём
delay(wait);
}
}

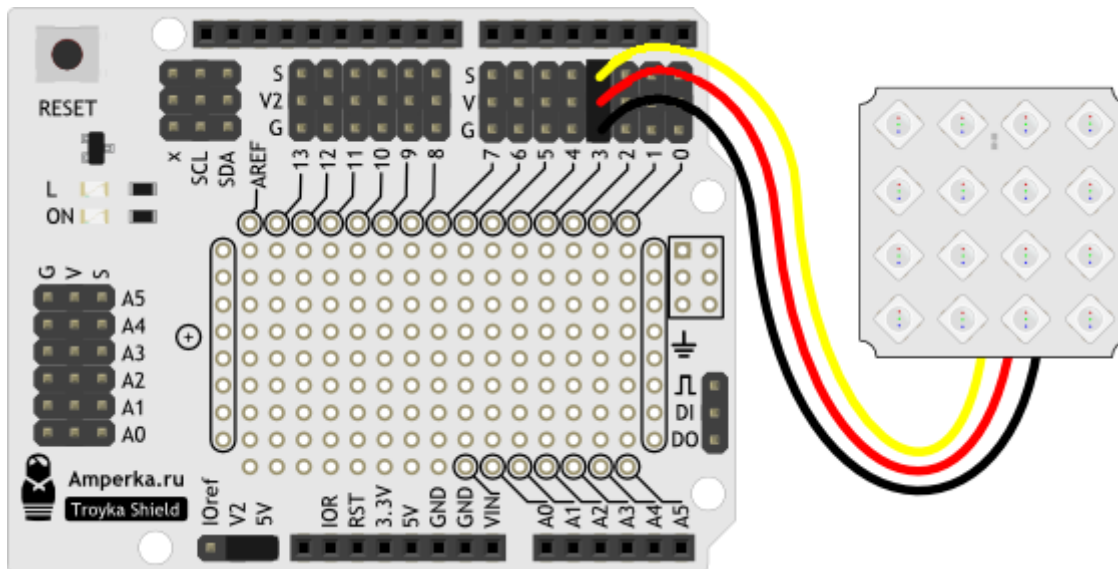
```

После прошивки управляющей платформы, вы увидите заполнение по очереди каждого светодиода матрицы из красного, зелёного и синего цветов.



## Пример работы для Iskra JS

Iskra JS использует для работы с модулем пин SPI MOSI, поэтому подключать модуль следует к пину P3:



```

SPI1.setup({baud:3200000, mosi:P3, sck:A5, miso:P2});
var length = 16;
var matrix = require('@amperka/led-strip').connect(SPI1, length, 'GRB');
// гасим все светодиоды
matrix.clear();

setInterval(function() {
  for (var i = 0; i < length; i++) {
    // выбираем произвольную интенсивность красного цвета

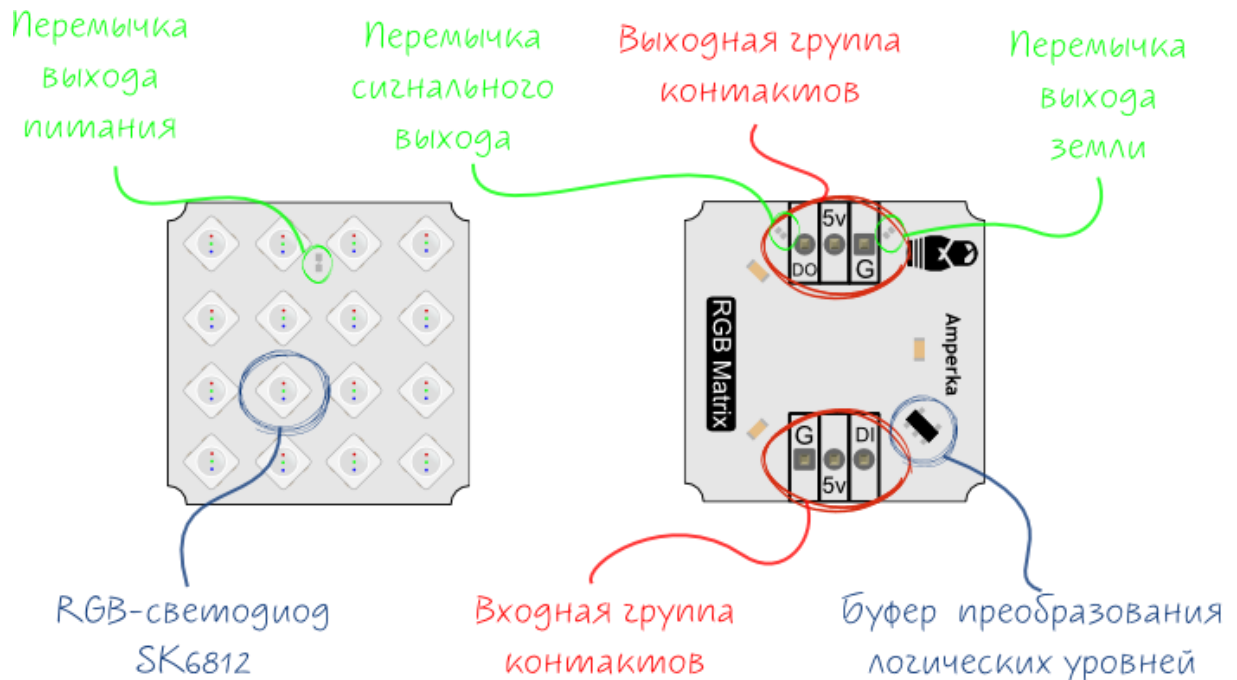
```

```

var redTone = Math.random();
// устанавливаем цвет светодиода
matrix.putColor(i, [redTone, 0, 0]);
}
// применяем изменения
matrix.apply();
}, 1000);

```

## Элементы платы



### RGB-светодиод SK6812

SK6812 представляет собой RGB-светодиод со встроенным контроллером. Протокол управления этим светодиодом аналогичен протоколу популярной микросхемы WS2812.

### Буфер преобразования логических уровней

Необходим для сопряжения устройств с разными напряжениями логических уровней. Родное напряжение модуля — 5 В, но он будет отлично работать и с Iskra JS, рабочее напряжение которой 3,3 В.

### Перемычки выходных пинов

Соедините капелькой припоя контакты специальных перемычек на модуле. Это позволит соединить несколько RGB-матриц последовательно.

Использовать модуль совместно с [Troyka Slot Shield](#) безопаснее с нераспаянными перемычками.

### Контакты подключения трёхпроводных шлейфов

*Входная группа контактов*

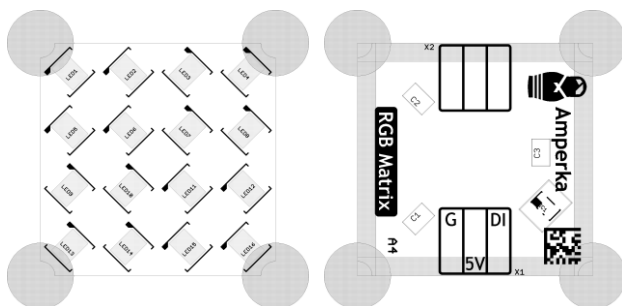
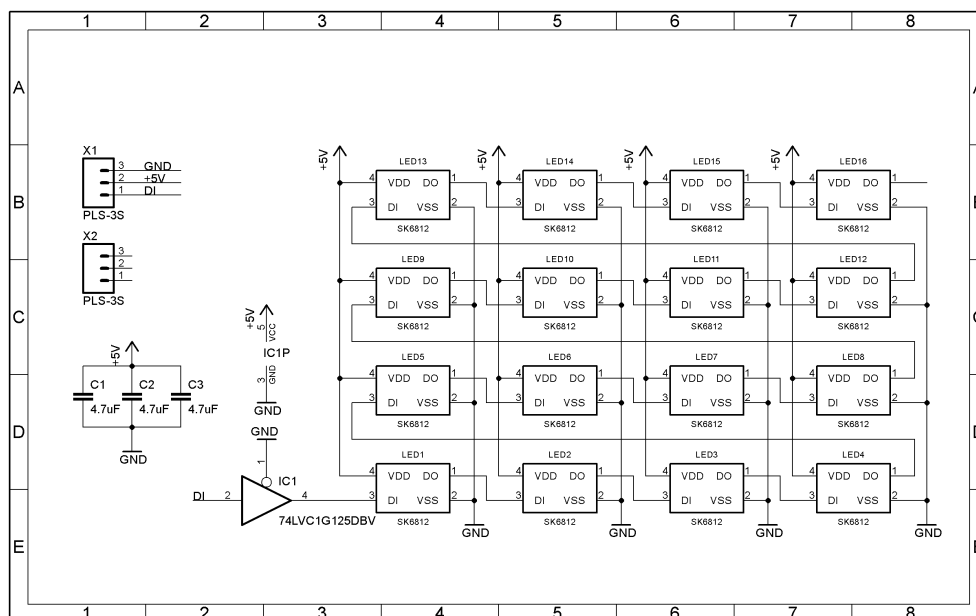
- Сигнальный (DI) — цифровой вход первого RGB-светодиода. Подключите к любому цифровому пину микроконтроллера.
- Питание (V) — питание модуля. Соедините с питанием микроконтроллера.
- Земля (G) — земля модуля. Соедините с землёй микроконтроллера.

### Выходная группа контактов

Используется, если требуется подключить несколько RGB-матриц в одну сигнальную линию. По умолчанию выключена. Для включения капните немного припоя на перемычки выходных пинов.

- Сигнальный (DO) — цифровой выход последнего RGB-светодиода.
- Сигнальный (V) — питание линии.
- Сигнальный (G) — земля линии.

## Принципиальная и монтажная схемы



## Характеристики

- Модуль: адресный RGB-светодиод SK6812
- Количество светодиодов: 16
- Рабочее напряжение: 3,3–5 В
- Потребляемый ток: до 0.6 А
- Габариты: 25,4×25,4 мм

## Ресурсы

- [Векторное изображение RGB-Matrix \(Top\)](#)
- [Векторное изображение RGB-Matrix \(Bottom\)](#)
- [Библиотека для Arduino](#)
- [Библиотека для Iskra JS](#)