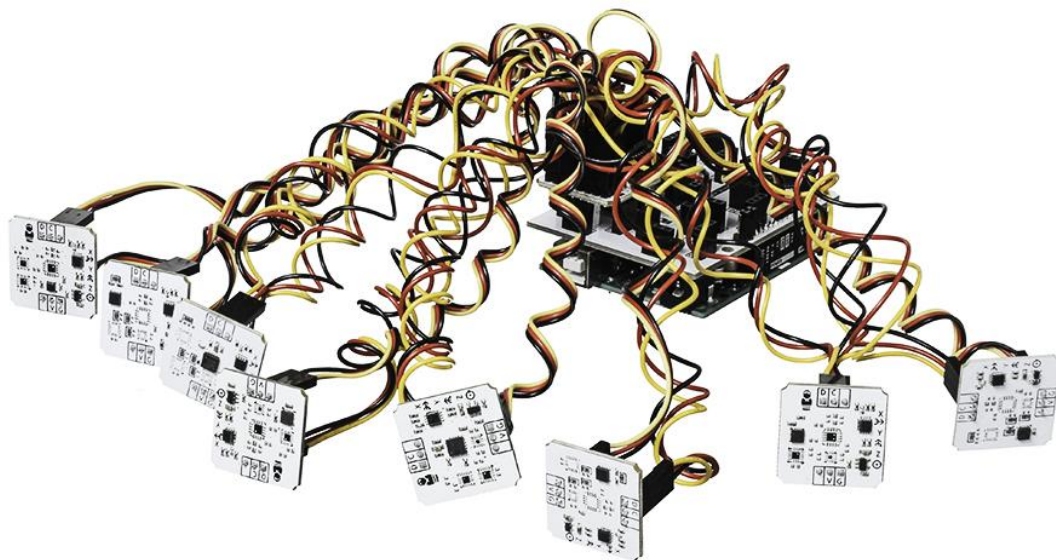


I²C хаб (Тройка-модуль)

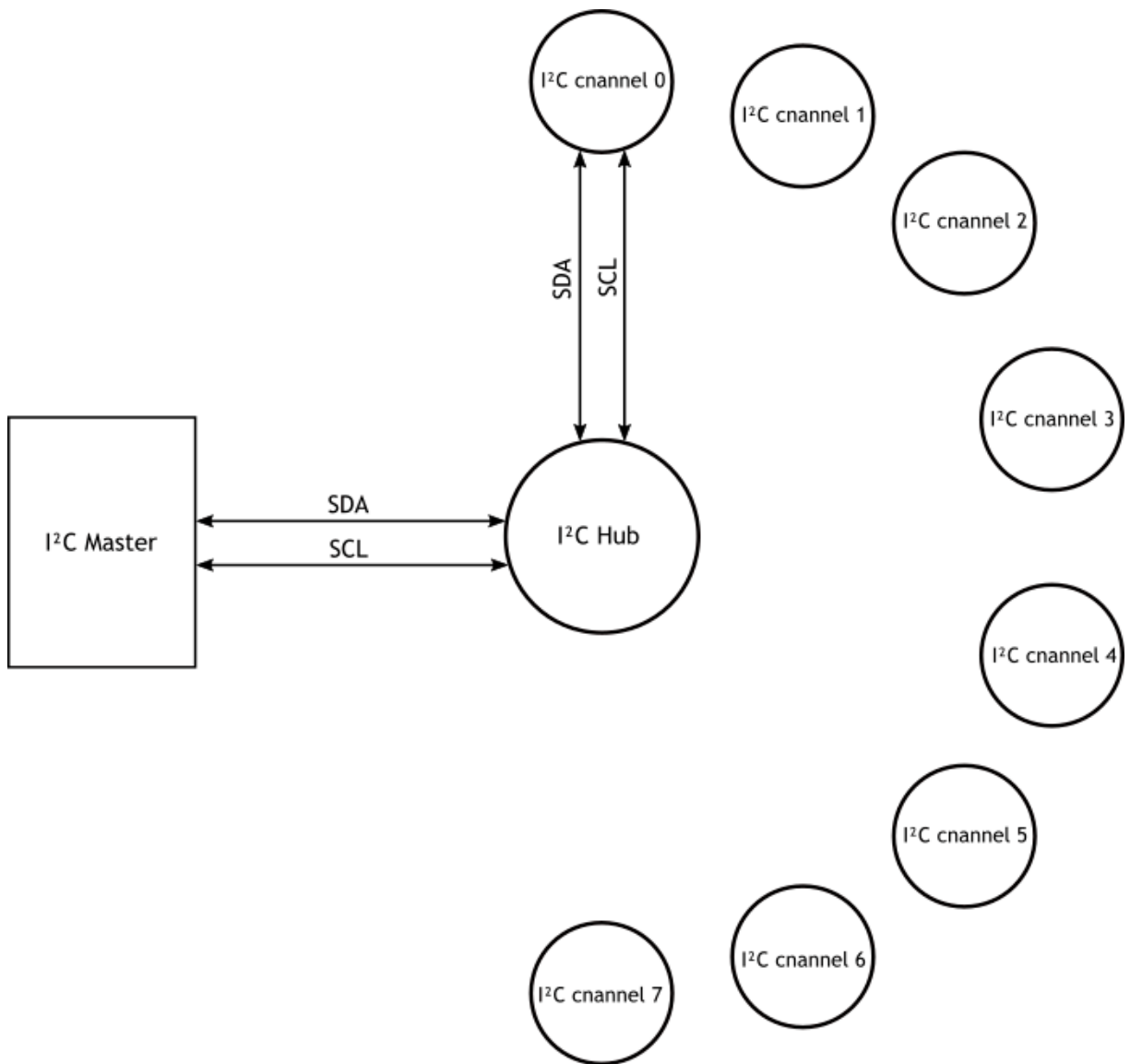
[I²C хаб](#) подружит модули с одинаковыми адресами.



Принцип работы

Работу модуля удобно представить в виде галетного переключателя.

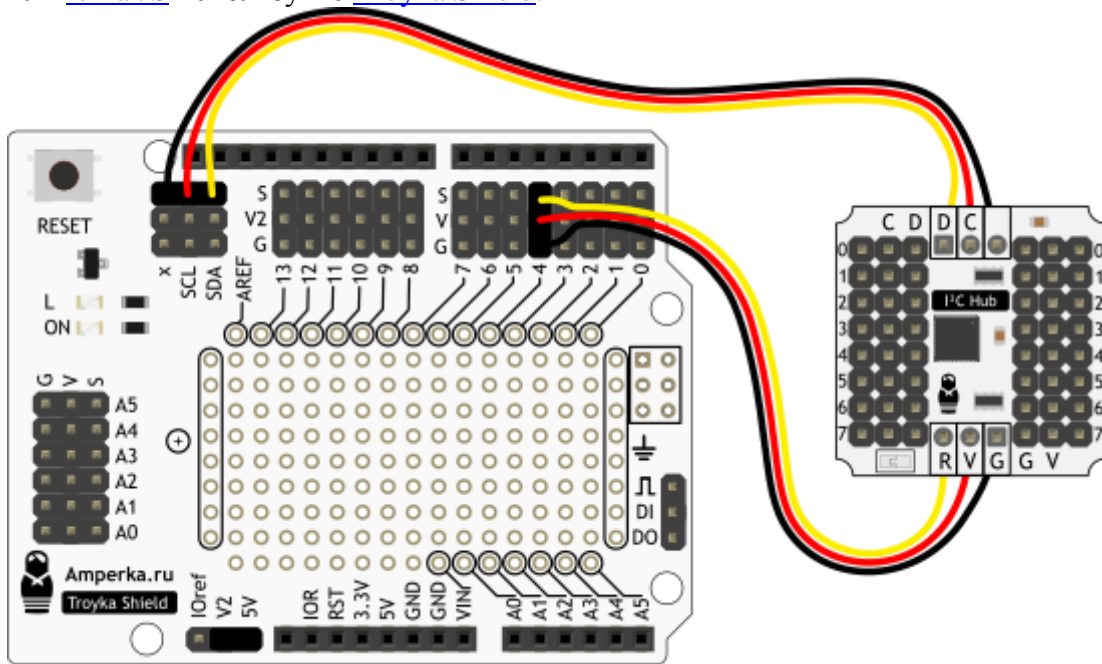
Есть управляющее устройство *Master* и подчинённые устройства *Slave* на восьми каналах I²C хаба. В каждый промежуток времени I²C хаб коммутирует *Master* только с одним устройством.



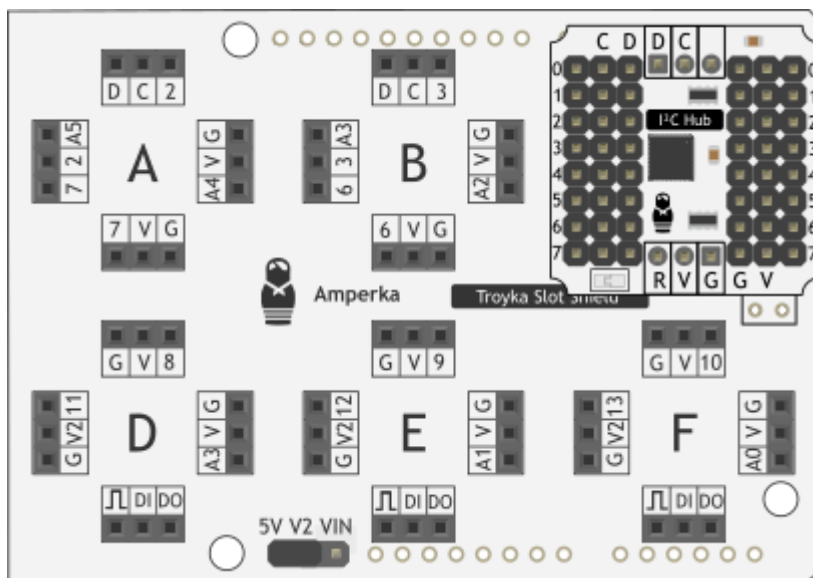
Хаб позволяет подключить к управляющей плате восемь I²C устройств с одинаковым адресом и опрашивать их по очереди, переключая каналы.

Подключение и настройка

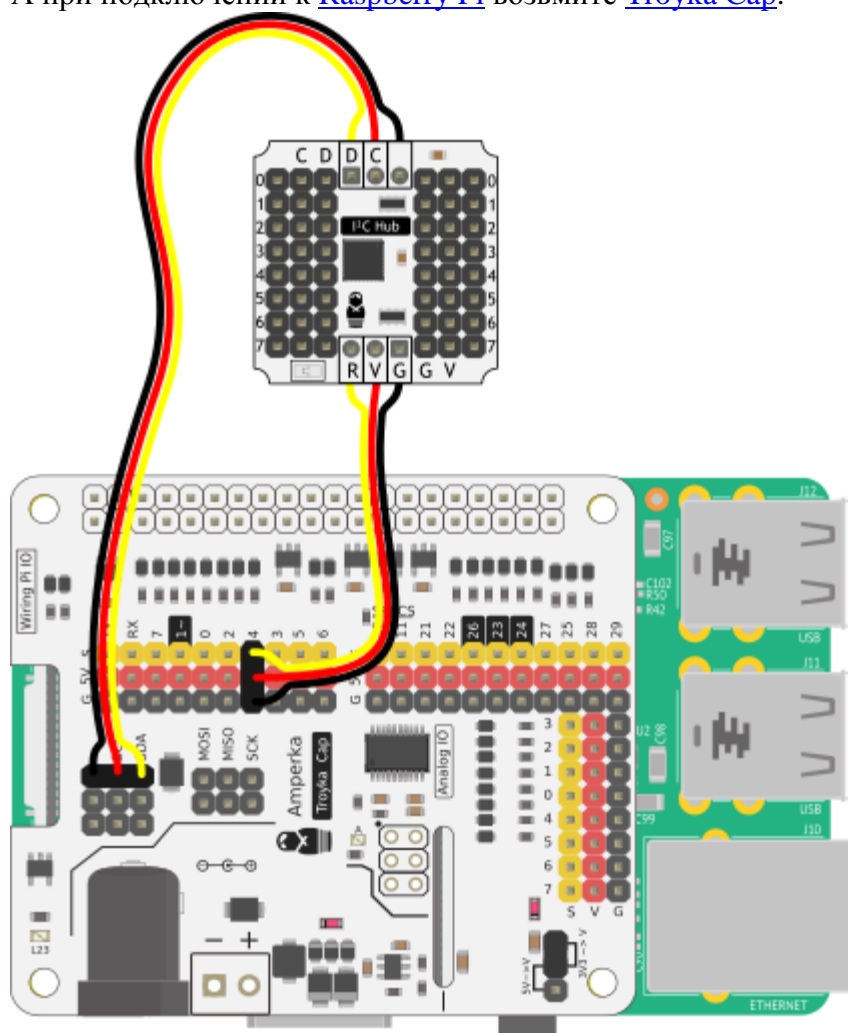
PC хаб общается с управляющей электроникой по [PC/TWI](#). При подключении к [Arduino](#) или [Iskra JS](#) используйте [Troyka Shield](#).



С [Troyka Slot Shield](#) провода не понадобятся вовсе.



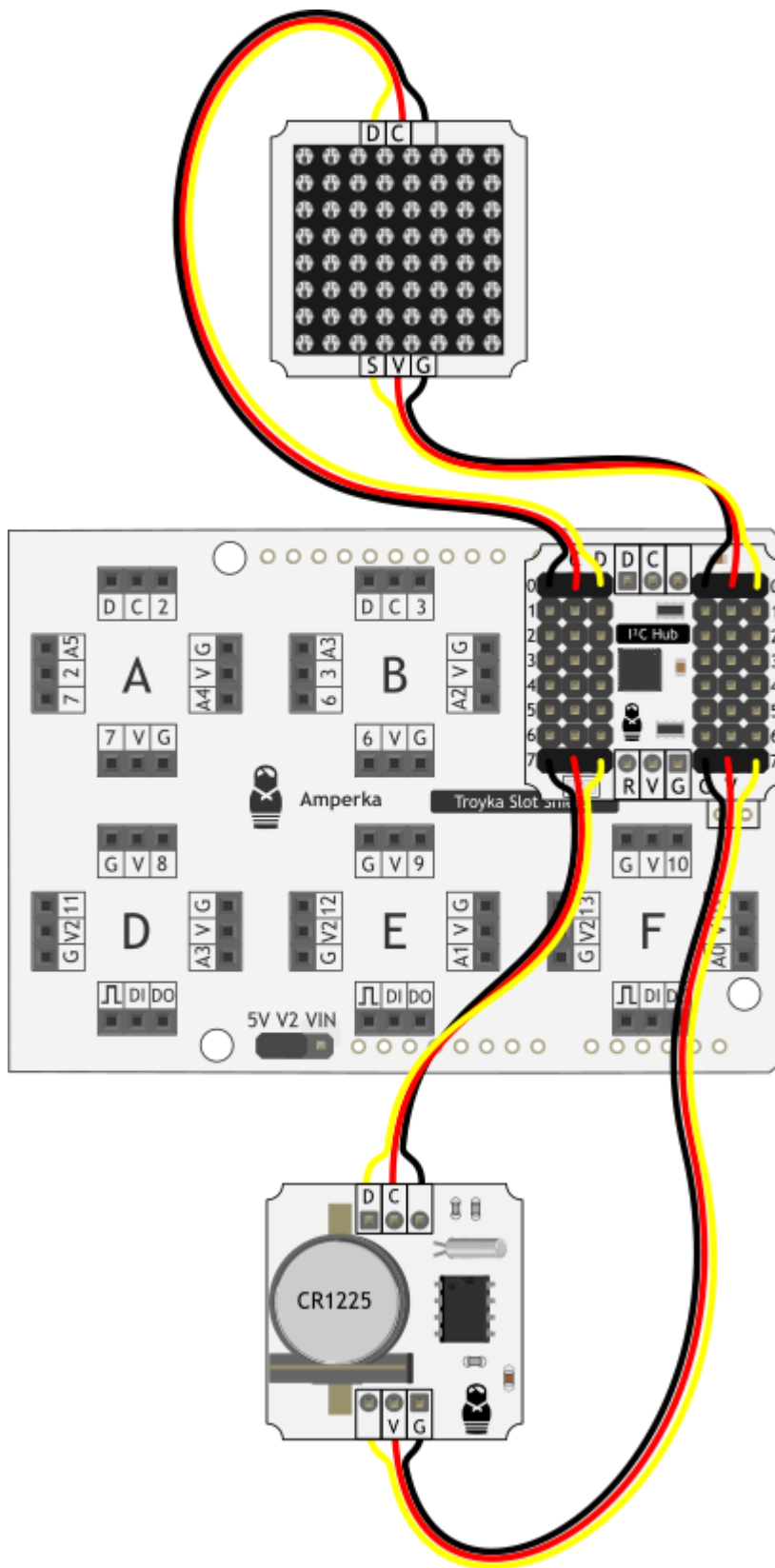
А при подключении к [Raspberry Pi](#) возьмите [Troyka Cap](#).



Пример работы

Подключим [часы реального времени](#) и [монохромную светодиодную матрицу](#) к Arduino через [I²C разветвитель](#).

Схема устройства



Код для Arduino

[i2cHubScanner.ino](#)

```
// библиотека для работы с I2C хабом
#include <ТройкаI2CHub.h>

// создаём объект для работы с хабом
// адрес по умолчанию 0x70
```

```

TroykaI2CHub splitter;

// можно создавать несколько объектов с разными адресами
// подробнее читайте на:
// http://wiki.amperka.ru/продукты:troyka-i2c-hub

void setup() {
  // открываем последовательный порт
  Serial.begin(115200);
  // ждём открытия порта
  while(!Serial) {
  }
  // печатаем сообщение об успешной инициализации Serial-порта
  Serial.println("Serial init OK");
  // начало работы с I2C хабом
  splitter.begin();
  Serial.println("Splitter init OK");
  // ждём одну секунду
  delay(1000);
}

void loop() {
  // счётчик цикла
  for (int i = 0; i < 8; i++) {
    // переключаем по очереди каналы
    splitter.setBusChannel(i);
    // выводим номер канала
    Serial.print("Set channel ");
    Serial.print(i);
    Serial.println(":");
    // запускаем I2C сканер
    startScannerI2C();
    // ждём одну секунду
    delay(1000);
  }
}

void startScannerI2C()
{
  // переменная состояние ответа
  byte state;
  // переменная хранения текущего адреса
  byte address;
  // переменная для хранения количества найденных I2C устройств
  int countDevices = 0;
  // печатем о начале поиска
  Serial.println("Scanning...");
  // перебираем по очереди все адреса от 0 до 127
  for (address = 1; address < 127; address++ ) {
    // начинаем передачу данных по текущем адресу
    Wire.beginTransmission(address);
    // завершаем передачу данных
    state = Wire.endTransmission();
    // если пришедший байт равен нулю
    if (state == 0) {
      // на адресе есть устройство
      // печатаем об этом
      Serial.print("I2C device found at address 0x");
      // если адрес меньше 16, печатем ноль
      if (address < 16) {
        Serial.print("0");
      }
      // печатаем текущий адрес в 16 разрядной системе исчисления
      Serial.print(address, HEX);
    }
  }
}

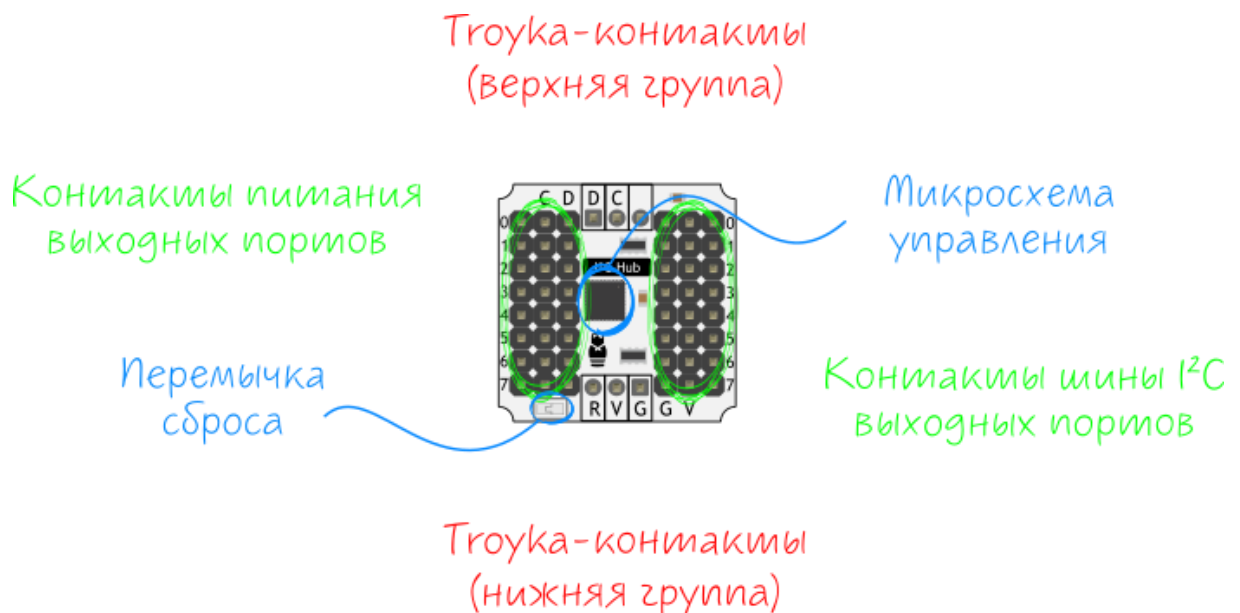
```

```

Serial.println(" !");
// инкрементируем кол-во найденных устройств
countDevices++;
}
}
// если не найдено ни одного I2C устройства
// печатаем об этом
if (countDevices == 0) {
Serial.println("No I2C devices found");
} else {
// печатаем о завершении процесса
Serial.println("Done");
}
}
}

```

Элементы платы



Микросхема управления

Сердце модуля — микросхема хаб [PCA9547](#) для модулей с шиной I²C.

Тройка-контакты

На модуле выведено две пары Тройка-контактов.

Нижняя группа

- Сигнальный (R) — сигнал сброса микросхемы управления PCA9547. Подключите к цифровому пину микроконтроллера. [По умолчанию пин отключён от чипа](#)
- Питание (V) — соедините с рабочим напряжением микроконтроллера.
- Земля (G) — Соедините с пином GND микроконтроллера.

Верхняя группа

- Сигнальный (D) — пин данных шины I²C. Подключите к SDA пину микроконтроллера.

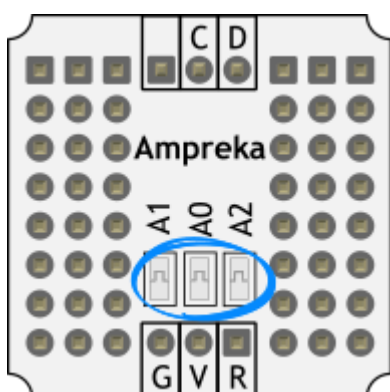
- Сигнальный (C) — пин тактирования шины I²C. Подключите к SCL пину микроконтроллера.

Контакты питания портов выхода

контакты шины I²C портов выхода

Площадки выбора адреса

Подключайте до восьми модулей к одному контроллеру путём смены адреса чипа. Для этого на обратной стороне модуля мы расположили три контактных площадки. Для смены адреса замкните интересующие контактные площадки с помощью [паяльника](#) и [припоя](#).

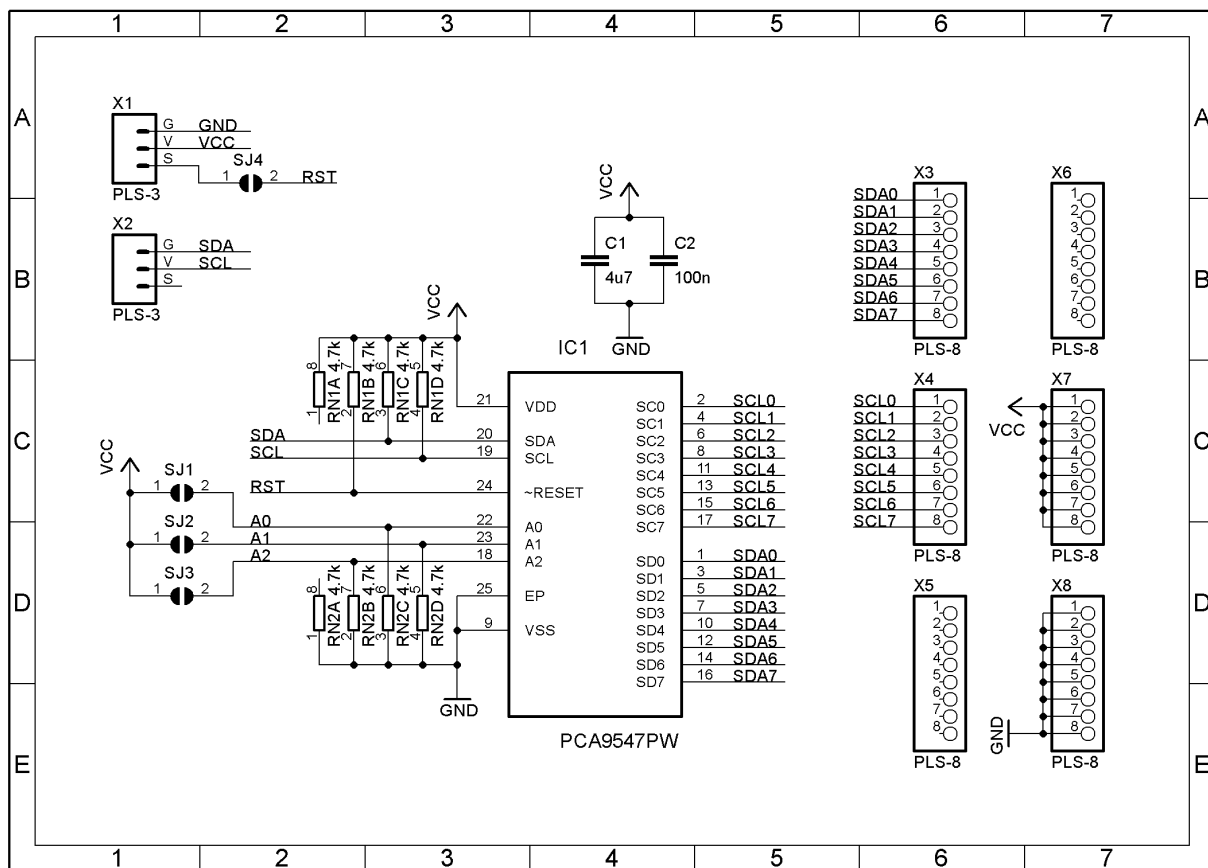


Адрес	A2	A1	A0
0x70	0	0	0
0x71	0	0	1
0x72	0	1	0
0x73	0	1	1
0x74	1	0	0
0x75	1	0	1
0x76	1	1	0
0x78	1	1	1

Переключатель сброса

По умолчанию ножка RESET микросхемы управления PCA9547 физически отключена от пина R. Для включения программного сброса капните каплей припоя на контактную площадку. После проделанных действий при подаче низкого уровня на пин R, микросхема перезагрузится.

Принципиальная схемы



Характеристики

- Напряжение питания: 3.3–5 В
- Интерфейс на вход: I²C
- Адрес модуля по умолчанию: 0x70
- Интерфейс на выход: восемь дополнительных каналов I²C
- Габариты: 25,4×25,4 мм

Ресурсы

- [Библиотека для Arduino](#)
- [Векторное изображение модуля \(TOP\)](#)
- [Векторное изображение модуля \(BOTTOM\)](#)
- [Datasheet на микросхему PCA9547](#)