

Практическое занятие №

Тема: «Подключение и программирование кнопочной матрицы с RGB-светодиодом»

Цель работы: приобрести практические навыки по подключению и программированию кнопочных массивов и многоцветных светодиодов на платформе Arduino.

Последовательность выполнения работы:

- Собрать схемы на макетной плате, иначе при отсутствии набора Arduino в web-приложениях (<https://wokwi.com/projects/new/arduino-uno> или <https://www.tinkercad.com/>) для приведенных примеров.
- Запрограммировать микроконтроллер согласно заданию в примере.

Содержание отчета:

- название практического занятия, его цель;
- фото или скриншоты собранной схемы;
- написанный программный код вставить текстом;
- вывод о проделанной работе;
- файл Fritzing с принципиальной и монтажной схемой.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

МАТРИЧНАЯ КЛАВИАТУРА 4×4

Принцип работы

Матричная клавиатура – это способ организации множества кнопок в виде матрицы для экономии выводов микроконтроллера. Вместо подключения каждой кнопки к отдельному пину используется матричная структура:

- 4 строки (rows)
- 4 столбца (columns)
- 16 кнопок на пересечениях

Экономия выводов

Без матрицы для 16 кнопок потребовалось бы 16 выводов + общий провод. С матрицей нужно всего 8 выводов (4 строки + 4 столбца).

Алгоритм опроса матрицы:

1. **Поочередное активирование строк:**
 - на одну строку подается LOW (0V)
 - остальные строки остаются в HIGH (5V) или в высокоимпедансном состоянии

2. Чтение столбцов:

- проверяется состояние каждого столбца
- если в активной строке нажата кнопка, соответствующий столбец будет иметь LOW уровень

Активная строка + столбец с LOW = координаты нажатой кнопки

Механический дребезг – явление, когда контакты кнопки несколько раз замыкаются и размыкаются в течение миллисекунд после нажатия.

Методы борьбы:

- программные задержки
- аппаратные RC-фильтры
- цифровая фильтрация в коде

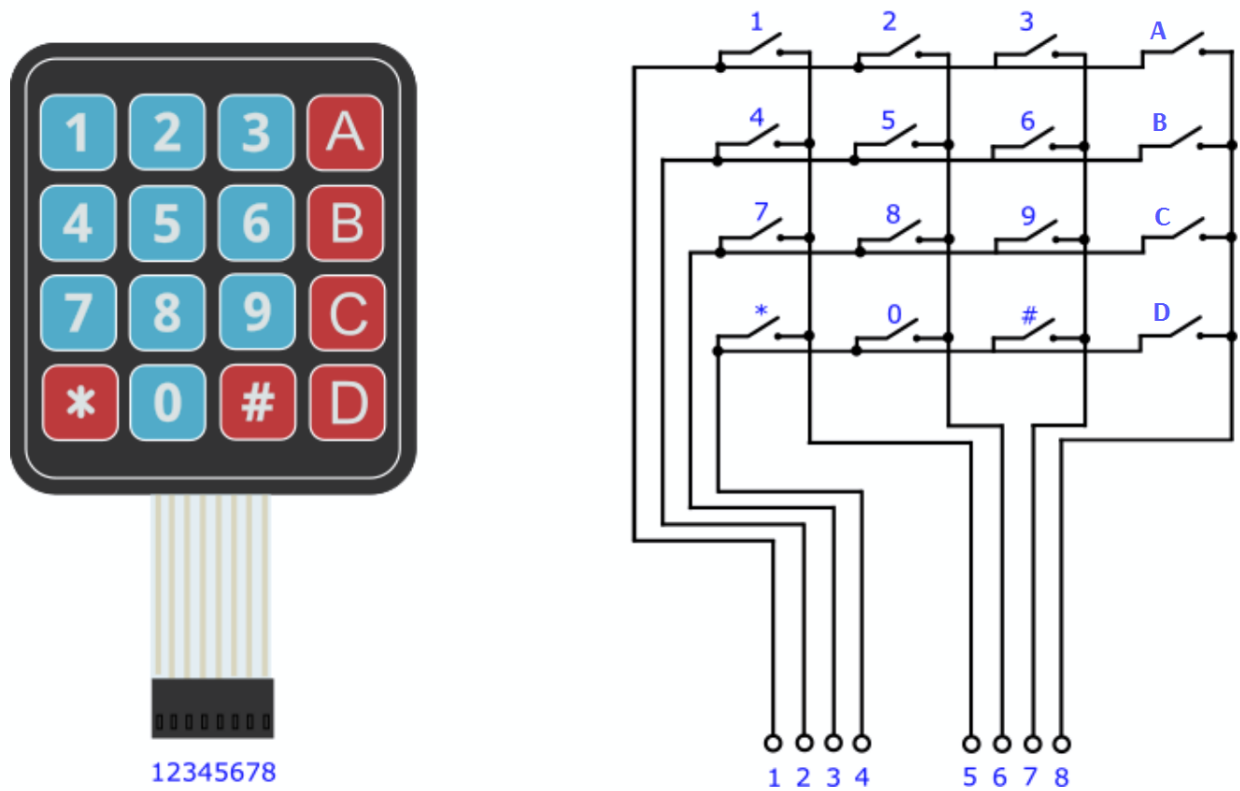


Рисунок 1 – Кнопочная матрица



Рисунок 2 – Кнопочная матрица с типичного набора

RGB-СВЕТОДИОД

RGB-светодиод содержит три независимых светодиодных кристалла в одном корпусе:

- R - красный (Red)
- G - зеленый (Green)
- B - синий (Blue)

ТИПЫ RGB-СВЕТОДИОДОВ

С общим катодом (Common Cathode)

Все три катода соединены вместе и подключаются к GND. Аноды управляются положительным напряжением, поэтому для включения цвета: подать HIGH или повышающее значение от 0 до 255 при использовании ШИМ на соответствующий пин.

С общим анодом (Common Anode)

Все три анода соединены вместе, подключены к VCC (+5V). Катоды управляются отрицательным напряжением, поэтому для включения цвета: подать LOW или повышающее значение от 255 до 0 при использовании ШИМ на соответствующий пин.

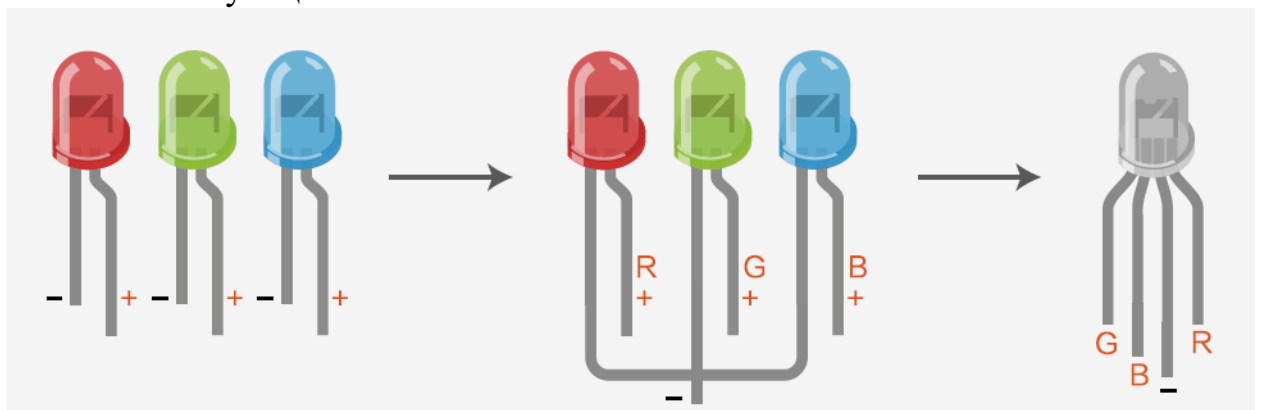
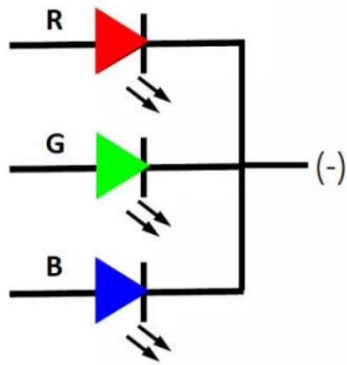


Рисунок 3 – Строение RGB-светодиода

Общий катод (-)



Общий анод (-)

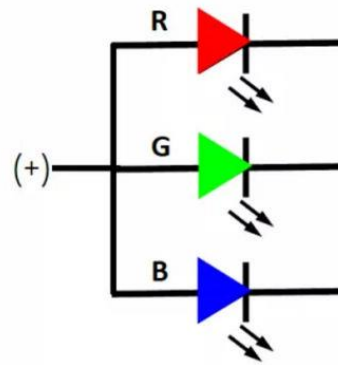


Рисунок 4 – Принцип подключения RGB-светодиода

ЗАДАНИЕ

Построить схему:

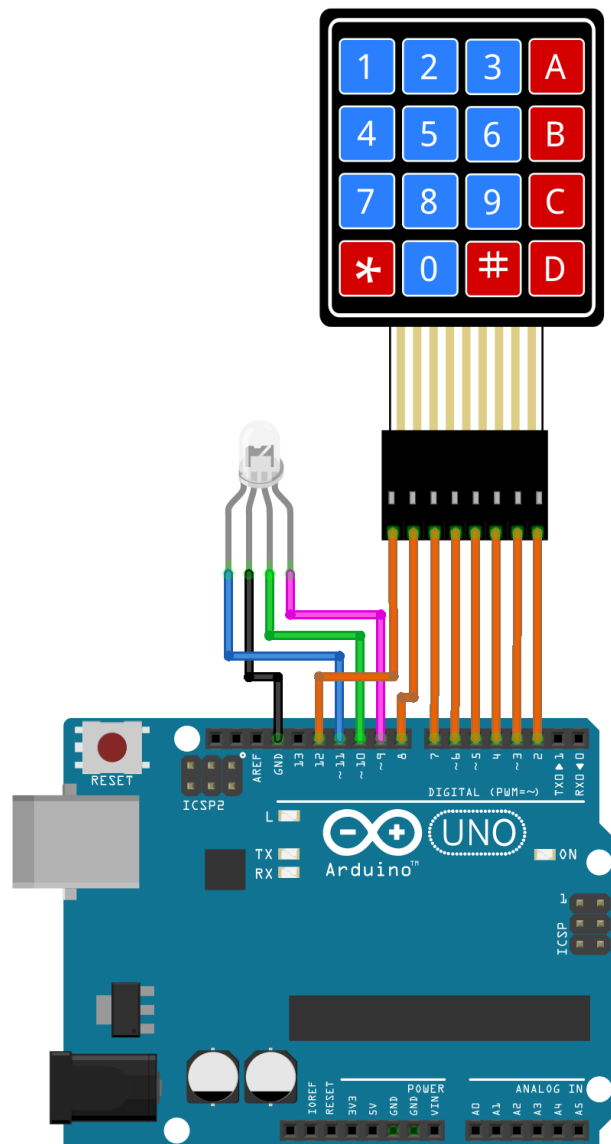


Рисунок 5 – Схема кодключения

Написать программу:

```
#include <Keypad.h>

// Определение количества пинов для матрицы кнопок
const byte ROWS = 4;
const byte COLS = 4;

// Пины для строк и столбцов матрицы
byte rowPins[ROWS] = {5, 4, 3, 2}; // Строки (4-1)
byte colPins[COLS] = {6, 7, 8, 12}; // Столбцы (1-4)

// Пины для RGB светодиода
const byte redPin = 9;
const byte greenPin = 10;
const byte bluePin = 11;

// Массив кнопок
char keys[ROWS][COLS] = {
  {'0', '1', '2', '3'},
  {'4', '5', '6', '7'},
  {'8', '9', 'A', 'B'},
  {'C', 'D', 'E', 'F'}
};

// Массив цветов остается без изменений
int colors[16][3] = {
  {255, 0, 0}, // 0: Красный
  {0, 255, 0}, // 1: Зеленый
  {0, 0, 255}, // 2: Синий
  {255, 255, 0}, // 3: Желтый
  {255, 0, 255}, // 4: Пурпурный
  {0, 255, 255}, // 5: Голубой
  {255, 165, 0}, // 6: Оранжевый
  {128, 0, 128}, // 7: Фиолетовый
  {255, 255, 255}, // 8: Белый
  {128, 128, 128}, // 9: Серый
  {255, 192, 203}, // 10: Розовый
  {0, 128, 0}, // 11: Темно-зеленый
  {64, 224, 208}, // 12: Бирюзовый
  {210, 105, 30}, // 13: Шоколадный
  {255, 215, 0}, // 14: Золотой
  {75, 0, 130} // 15: Индиго
};
```

```

Keypad keypad = Keypad(makeKeymap(keys), rowPins, colPins,
ROWS, COLS);

void setup() {
  Serial.begin(9600);
  pinMode(redPin, OUTPUT);
  pinMode(greenPin, OUTPUT);
  pinMode(bluePin, OUTPUT);
  setColor(0, 0, 0);
  Serial.println("Контроллер с кнопочной матрицей для RGB");
}

void loop() {
  char key = keypad.getKey();

  if (key) {
    int keyIndex = getKeyIndex(key);

    if (keyIndex != -1) {
      setColor(colors[keyIndex][0], colors[keyIndex][1],
colors[keyIndex][2]);
      Serial.print("Кнопка: ");
      Serial.print(key);
      Serial.print(" - Цвет: R=");
      Serial.print(colors[keyIndex][0]);
      Serial.print(" G=");
      Serial.print(colors[keyIndex][1]);
      Serial.print(" B=");
      Serial.println(colors[keyIndex][2]);
    } else {
      Serial.print("Неизвестная клавиша: ");
      Serial.println(key);
    }
  }
}

void setColor(int red, int green, int blue) {
  analogWrite(redPin, red);
  analogWrite(greenPin, green);
  analogWrite(bluePin, blue);
}

// Функция обработки символов A-F

```

```
int getKeyIndex(char key) {  
    // Преобразуем символы '0'-'9' в числа 0-9  
    if (key >= '0' && key <= '9') {  
        return key - '0';  
    }  
    // Преобразуем символы 'A'-'F' в числа 10-15  
    if (key >= 'A' && key <= 'F') {  
        return key - 'A' + 10;  
    }  
    return -1;  
}
```