



Ziel des Projekts

Ein LED-Streifen soll nach Betätigung einer Taste verschiedene Farben anzeigen. Die LEDs sollen in unterschiedlichen Reihenfolgen leuchten.

Die Hardware



Der LED-Streifen besteht aus mehreren miteinander verbundenen RGB-LEDs. Jede besitzt einen eigenen Controller und kann einzeln angesteuert werden. Er benötigt nur einen digitalen Eingang. LED-Streifen gibt es in verschiedenen Längen.

RGB ist eine Mischung der Farben Rot, Grün und Blau. Jede Farbe kann von 0 bis 255 gesetzt werden, die Werte werden durch Kommata getrennt.

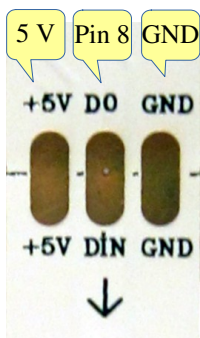
Beispiele:

163, 0, 93

226, 176, 50

255, 255, 0

255, 228, 225

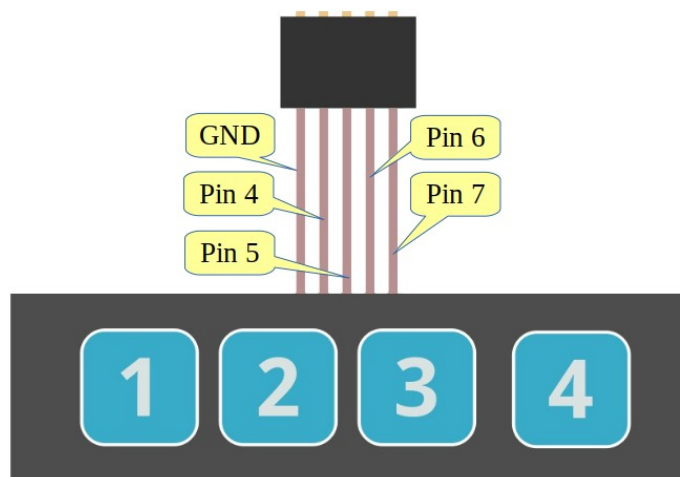


Jede LED kann einzeln angesprochen werden.

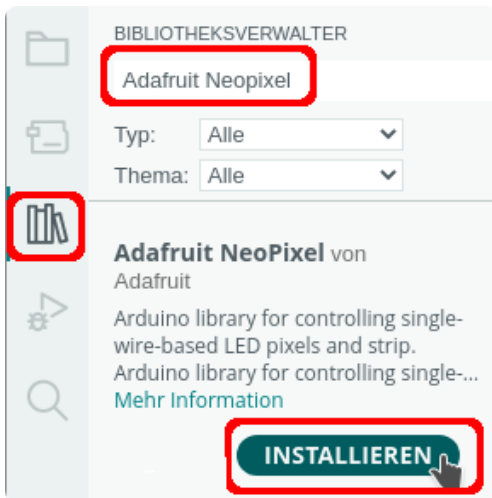
Die Zählung beginnt mit 0!

Benötigte Bauteile

- LED-Streifen
- Tastenfeld 1×4
- Leitungsdrähte

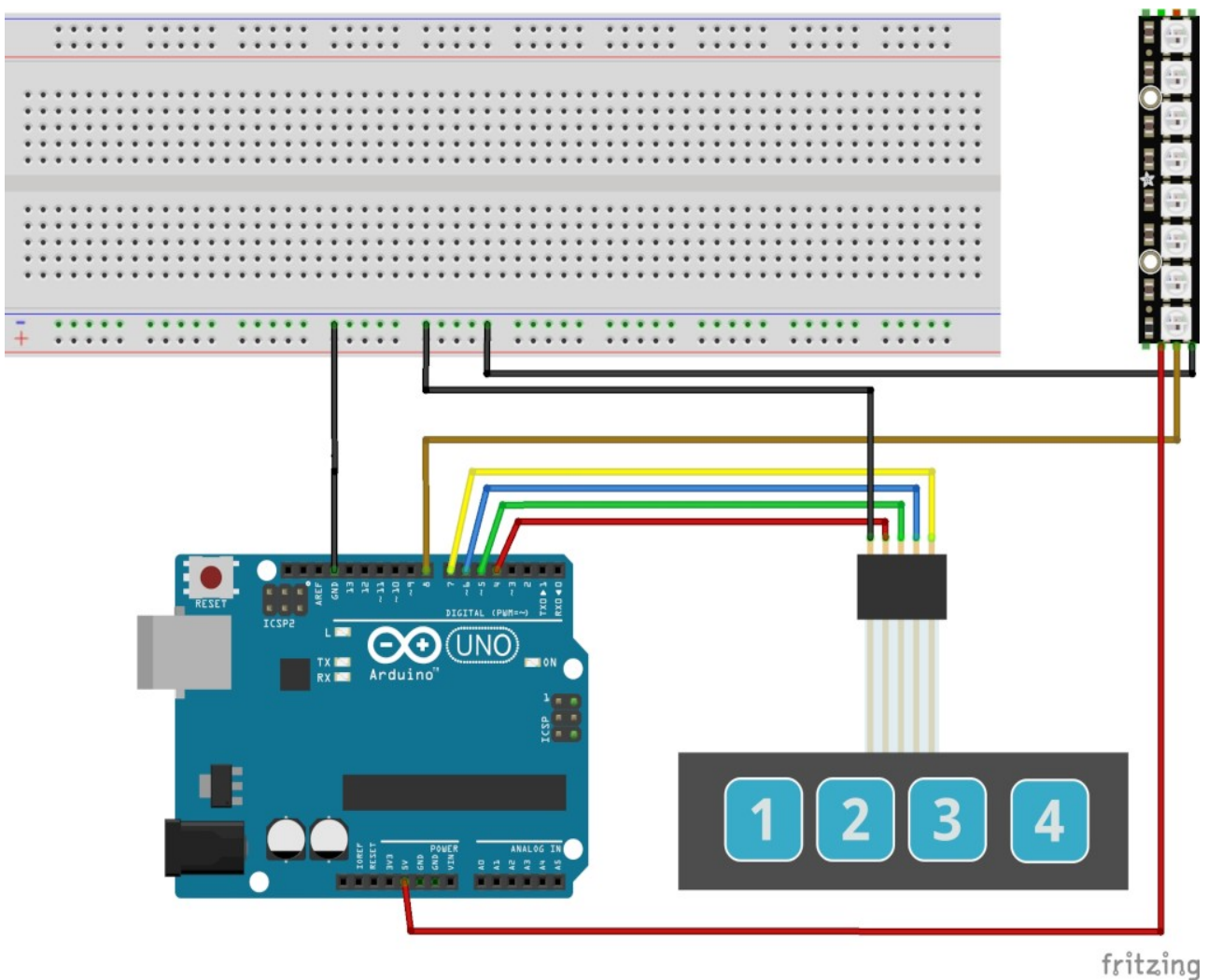


Bibliothek installieren



Sketch → Bibliothek einbinden → Bibliotheken verwalten

Der Schaltplan





Das Programm

Tastenfeld prüfen



Teste zunächst, ob die Tasten 1 bis 4 den richtigen Pins zugeordnet wurden. Die gedrückte Taste wird im Seriellen Monitor angezeigt.

```
int TASTER_EINS = 5; // Taste 1
int TASTER_ZWEI = 4; // Taste 2
int TASTER_DREI = 7; // Taste 3
int TASTER_VIER = 6; // Taste 4

void setup()
{
  Serial.begin(9600);

  // Vorwiderstand aktivieren
  pinMode(TASTER_EINS, INPUT_PULLUP);
  pinMode(TASTER_ZWEI, INPUT_PULLUP);
  pinMode(TASTER_DREI, INPUT_PULLUP);
  pinMode(TASTER_VIER, INPUT_PULLUP);
}

void loop()
{
  if (digitalRead(TASTER_EINS) == LOW)
  {
    delay(100);
    Serial.println("Taste 1 an Pin " + String(TASTER_EINS));
  }

  if (digitalRead(TASTER_ZWEI) == LOW)
  {
    delay(100);
    Serial.println("Taste 2 an Pin " + String(TASTER_ZWEI));
  }

  if (digitalRead(TASTER_DREI) == LOW)
  {
    delay(100);
    Serial.println("Taste 3 an Pin " + String(TASTER_DREI));
  }

  if (digitalRead(TASTER_VIER) == LOW)
  {
    delay(100);
    Serial.println("Taste 4 an Pin " + String(TASTER_VIER));
  }
}
```



Funktionen der Bibliothek Adafruit_NeoPixel (Auswahl)

Schlüsselwort	Aktion
begin()	LED-Streifen starten
numPixels()	Anzahl der LEDs lesen
show()	LED-Streifen einschalten
clear()	LED-Streifen ausschalten
setPixelColor(Nummer, rot, grün, blau)	Farbe einer LED setzen Nummer → Nummer der LED rot -> 0 - 255 grün -> 0 - 255 blau -> 0 - 255
setBrightness()	Helligkeit setzen (0-255)
Color(rot, grün, blau)	Farbe für alle LEDs setzen rot -> 0 - 255 grün -> 0 - 255 blau -> 0 - 255 Beispiel blau: int Farbe = LEDRing.Color(0, 0, 255);
fill(Farbe, Start, Ende)	Farbe für die mit Start und Ende bezeichneten Pixel setzen



Beispiele

Farbwechsel

```
#include "Adafruit_NeoPixel.h"
#define STREIFEN 8

// Anzahl der LEDs -> muss angepasst werden
#define AnzahlLED 15

// LED-Streifen -> Name des LED-Streifens
Adafruit_NeoPixel LEDStreifen = Adafruit_NeoPixel(AnzahlLED, STREIFEN, NEO_GRB + NEO_KHZ800);

void setup()
{
    // setBrightness(0..255)
    LEDStreifen.setBrightness(200);

    // NeoPixel Bibliothek initialisieren
    LEDStreifen.begin();
}

void loop()
{
    LEDStreifen.clear();
    LEDStreifen.show();

    int Farbe;

    // rot
    Farbe = LEDStreifen.Color(255, 0, 0);
    LEDStreifen.fill(Farbe, 0, AnzahlLED);
    LEDStreifen.show();
    delay(1000);

    // grün
    Farbe = LEDStreifen.Color(0, 255, 0);
    LEDStreifen.fill(Farbe, 0, AnzahlLED);
    LEDStreifen.show();
    delay(1000);

    // blau
    Farbe = LEDStreifen.Color(0, 0, 255);
    LEDStreifen.fill(Farbe, 0, AnzahlLED);
    LEDStreifen.show();
    delay(1000);

    // gelb
    Farbe = LEDStreifen.Color(255, 255, 0);
    LEDStreifen.fill(Farbe, 0, AnzahlLED);
    LEDStreifen.show();
    delay(1000);
}
```



```
// pink
Farbe = LEDStreifen.Color(255, 20, 147);
LEDStreifen.fill(Farbe, 0, AnzahlLED);
LEDStreifen.show();
delay(1000);

// Pause
LEDStreifen.clear();
LEDStreifen.show();
delay(2000);
}

// Farbe rot dimmen
# include <Adafruit_NeoPixel.h>
int STREIFEN = 8;

// Anzahl der LEDs -> muss angepasst werden
int AnzahlLED = 15;

/*
  Initialisierung des LED-Streifens
  Parameter:
  LEDStreifen -> Name des LED-Streifens
  AnzahlLED -> Anzahl der LEDs
  STREIFEN -> verwendeter Pin
  NEO_GRB + NEO_KHZ800 -> Typ des verwendeten LED-Streifens
*/
Adafruit_NeoPixel LEDStreifen = Adafruit_NeoPixel(AnzahlLED, STREIFEN, NEO_GRB + NEO_KHZ800);

void setup()
{
  // NeoPixel Bibliothek initialisieren
  LEDStreifen.begin();

  // Helligkeit setzen (0 ... 255)
  LEDStreifen.setBrightness(200);
}

void loop()
{
  /* rot dimmen
    die Variable i erhöht den Anteil von rot
    von 0 bis 255
    LEDNummer -> Nummer der LED
    0 bis numPixels() = Anzahl der LEDs im Streifen
    LEDStreifen.setPixelColor(LEDNummer, LEDStreifen.Color(i, 0, 0));
    -> Farbe der LED (LEDNummer) setzen
  */
```



```
for (int i = 0; i < 255; i++)
{
    for (int LEDNummer = 0; LEDNummer < LEDStreifen.numPixels(); LEDNummer++)
    {
        LEDStreifen.setPixelColor(LEDNummer, LEDStreifen.Color(i, 0, 0));
        LEDStreifen.show();
    }
}

// Anteil der Farbe verringern, dadurch entsteht der "Dimm-Effekt"
for (int i = 255; i > 0; i--)
{
    for (int LEDNummer = 0; LEDNummer < LEDStreifen.numPixels(); LEDNummer++)
    {
        LEDStreifen.setPixelColor(LEDNummer, LEDStreifen.Color(i, 0, 0));
        LEDStreifen.show();
    }
}
// LED aus
LEDStreifen.clear();
LEDStreifen.show();
delay(1000);
}
```

Lauflicht

```
#include "Adafruit_NeoPixel.h"
#define STREIFEN 8

// Anzahl der LEDs -> muss angepasst werden
#define AnzahlLED 15

// Taste 1 wird verwendet
int TASTER = 5;

// Anzahl der LEDs -> muss angepasst werden
int AnzahlLED = 15;

// LEDStreifen -> Name des LED-Streifens
Adafruit_NeoPixel LEDStreifen = Adafruit_NeoPixel(AnzahlLED, STREIFEN, NEO_GRB + NEO_KHZ800);

void setup()
{
    // NeoPixel Bibliothek initialisieren
    LEDStreifen.begin();
    LEDStreifen.setBrightness(200);
    pinMode(TASTER, INPUT_PULLUP);
}

void loop()
{
    int TasterLesen = digitalRead(TASTER);
```



```
// LEDs leuchten solange der Taster gedrückt wird
if (TasterLesen == LOW)
{
    int WarteZeit = 100;
    for (int q = 0; q < 3; q++)
    {
        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED mit blau einschalten
            LEDStreifen.setPixelColor(i + q, 0, 0, 255);
        }
        LEDStreifen.show();
        delay(WarteZeit);
        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED ausschalten
            LEDStreifen.setPixelColor(i + q, 0);
        }
    }
}

else
{
    // alle LEDs ausschalten
    LEDStreifen.clear();
    LEDStreifen.show();
}

else
{
    // alle LEDs ausschalten
    LEDStreifen.clear();
    LEDStreifen.show();
}
}
```

Jagdeffekt

Das Ein- und Ausschalten jeder dritten LED ergibt eine Art „Jagdeffekt“ der LEDs. Jede Taste des Tastenfeldes soll diesen Effekt bei der jeweiligen Farbe starten.

Binde die benötigte Bibliothek ein und definiere die Variablen:

```
#include "Adafruit_NeoPixel.h"
#define STREIFEN 8

// Anzahl der LEDs -> muss angepasst werden
#define AnzahlLED 15

// die Tasten
int TASTER_ROT = 5; // Taste 1
int TASTER_GRUEN = 4; // Taste 2
```



```
int TASTER_BLAU = 7; // Taste 3
int TASTER_GELB = 6; // Taste 4

// LEDStreifen -> Name des LED-Streifens
Adafruit_NeoPixel LEDStreifen = Adafruit_NeoPixel(AnzahlLED, STREIFEN, NEO_GRB + NEO_KHZ800);
```

Im setup-Teil wird der LED-Streifen gestartet und die Tasten des Tastenfelds als INPUT_PULLUP definiert.

```
void setup()
{
  // NeoPixel Bibliothek initialisieren
  LEDStreifen.begin();

  LEDStreifen.setBrightness(200);

  // Vorwiderstand aktivieren
  pinMode(TASTER_ROT, INPUT_PULLUP);
  pinMode(TASTER_GRUEN, INPUT_PULLUP);
  pinMode(TASTER_BLAU, INPUT_PULLUP);
  pinMode(TASTER_GELB, INPUT_PULLUP);
}
```

Im loop-Teil werden die Tasten einzeln abgefragt:

```
void loop()
{
  // alle LEDs ausschalten
  LEDStreifen.clear();
  LEDStreifen.show();

  // Taste 1
  if (digitalRead(TASTER_ROT) == LOW)
  {
    int WarteZeit = 100;

    for (int ii = 0; ii < 3; ii++)
    {
      for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
      {
        // jede dritte LED einschalten
        LEDStreifen.setPixelColor(i + ii, 200, 0, 0);
      }

      LEDStreifen.show();
      delay(WarteZeit);

      for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
      {
        // jede dritte LED ausschalten
        LEDStreifen.setPixelColor(i + ii, 0);
      }
    }
  }
}
```



```
// Taste 2
if (digitalRead(TASTER_GRUEN) == LOW)
{
    int WarteZeit = 100;

    for (int ii = 0; ii < 3; ii++)
    {
        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED einschalten
            LEDStreifen.setPixelColor(i + ii, 0, 255, 0);
        }

        LEDStreifen.show();
        delay(WarteZeit);

        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED ausschalten
            LEDStreifen.setPixelColor(i + ii, 0);
        }
    }
}

// Taste 3
if (digitalRead(TASTER_BLAU) == LOW)
{
    int WarteZeit = 100;

    for (int ii = 0; ii < 3; ii++)
    {
        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED einschalten
            LEDStreifen.setPixelColor(i + ii, 0, 0, 255);
        }

        LEDStreifen.show();
        delay(WarteZeit);

        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED ausschalten
            LEDStreifen.setPixelColor(i + ii, 0);
        }
    }
}
```



```
// Taste 4
if (digitalRead(TASTER_GELB) == LOW)
{
    int WarteZeit = 100;
    for (int ii = 0; ii < 3; ii++)
    {
        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED einschalten
            // zufällige Farbe
            LEDStreifen.setPixelColor(i + ii, 255, 255, 0);
        }

        LEDStreifen.show();
        delay(WarteZeit);

        for (int i = 0; i < LEDStreifen.numPixels(); i += 3)
        {
            // jede dritte LED ausschalten
            LEDStreifen.setPixelColor(i + ii, 0);
        }
    }
}
```

Hartmut Waller letzte Änderung: 22.11.25