

Halloween-Kürbis mit Arduino



Halloween-Kürbisse finde ich einfach super. In diesem Projekt zeige ich Dir, wie Du einen Halloween-Kürbis mit Arduino bauen kannst, der mit seinen Augen hin und her guckt. Dazu nutzen wir einen WS2812 LED-Strip.

Ausführliche Infos zum WS2812 kannst du gern noch mal hier nachlesen:

[Viele LEDs mit Arduino steuern – WS2812](#)

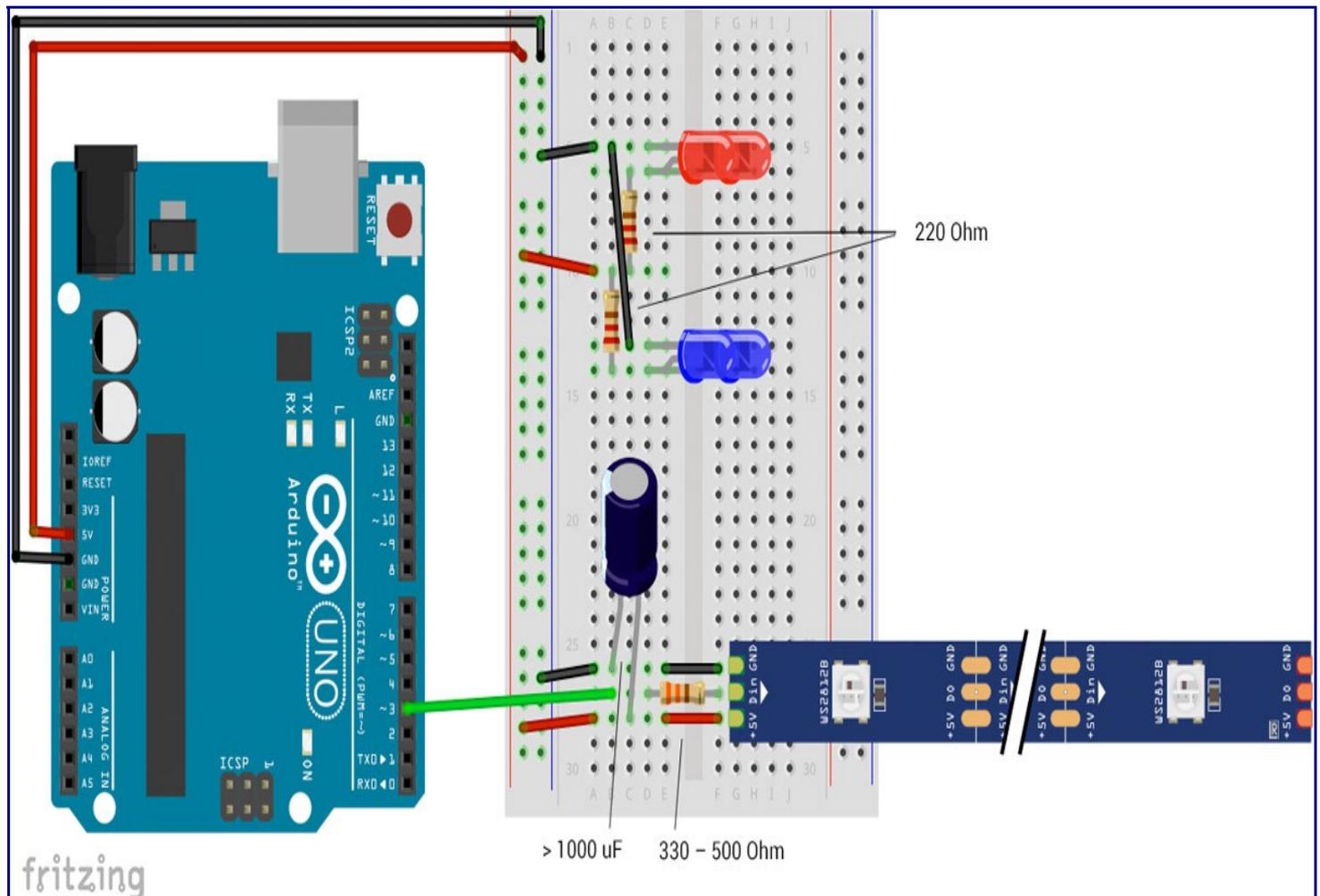
Bauteile

- [Arduino-Starterkit](#)
- [WS2812*](#) LED-Strip (kurz, 17 LEDs hab ich verwendet)
- Kondensator (ab 1000uF)
- Widerstand 220 Ohm für LEDs (rot, rot, braun bzw. rot-rot-schwarz-schwarz)
- Widerstand 330 – 500 Ohm
- 4 LEDs (ich hab zwei blaue und zwei rote verwendet)

Verhalten des Halloween-Kürbis mit Arduino

Der Leuchtstreifen soll die Augen vom Kürbis darstellen. Er leuchtet überall schwach rot, hat aber zwei helle weiße Augen, die nervös hin und her gucken.

Schaltplan



Der Daten-In-Pin des WS2812 LED-Strip ist über einen Widerstand von 330 Ohm (bis 500 Ohm sollte auch ok sein) mit dem Digitalen Pin 3 des Arduino-Boards verbunden. Plus und GND vom Strip sind direkt an den Arduino 5V+

<https://starthardware.org/halloween-kuerbis-mit-arduino/>

und GND angeschlossen. Dazu sind zwei rote LEDs und zwei blaue LEDs jeweils über einen 220 Ohm Widerstand mit dem 5V+ und GND verbunden. Sie sollen den Halloween-Kürbis von innen ausleuchten.

Code

Der Code setzt auf die Adafruit NeoPixel Library auf. Wenn du die noch nicht installiert hast, kannst du das in der Arduino-Software machen. Klickt auf *Sketch*>*Bibliotheken einbinden*>*Bibliothek* verwalten. Schreibe NeoPixel in das Suchfeld und installiere die NeoPixel Library von Adafruit in ihrer aktuellen Version.

Die Methode `evilEyes()` wird im Loop aufgerufen und erhält unter Anderem ein Parameter für die neue Augenposition. Die Methode selbst animiert die Augen dann an die vorgegebene Position. Sie sorgt ebenfalls dafür, dass alle anderen LEDs schwach rot leuchten.

```
#include <Adafruit_NeoPixel.h>
#ifdef __AVR__
#include <avr/power.h> // Required for 16 MHz Adafruit Trinket
#endif

#define LED_PIN    3
#define LED_COUNT 17

int eyePosition = 4;
int eyeDistance = 5;

Adafruit_NeoPixel strip(LED_COUNT, LED_PIN, NEO_GRB + NEO_KHZ800);

void setup() {
  strip.begin();
  strip.show();
  strip.setBrightness(50); // Set BRIGHTNESS to about 1/5 (max = 255)
}

void loop() {
  delay(random(100, 1000));
  evilEyes(strip.Color(20, 0, 0), random(10, 25), random(11));
}

void evilEyes(uint32_t color, int wait, int newEyePosition) {

  while (newEyePosition != eyePosition) {
    if (newEyePosition < eyePosition) eyePosition--;
    if (newEyePosition > eyePosition) eyePosition++;

    for (int i = 0; i < strip.numPixels(); i++) { // For each pixel in strip...
      if (i == eyePosition) strip.setPixelColor(eyePosition, strip.Color(255, 255, 255));
      else if (i == eyePosition + eyeDistance) strip.setPixelColor(eyePosition + eyeDistance,
strip.Color(255, 255, 255));
      else {
        strip.setPixelColor(i, color);           // Set pixel's color (in RAM)
        strip.show();                             // Update strip to match
        //delay(wait);                             // Pause for a moment
      }
    }
    delay(wait);
  }
}
```

Jetzt muss das alles nur noch in den Kürbis. Ich empfehle dir, den Kürbis von innen mit einem Müllbeutel

<https://starthardware.org/halloween-kuerbis-mit-arduino/>

auszukleiden und die Elektronik dort hinein zu packen. Kürbisse verfaulen gerade bei unseren Temperaturen relativ schnell und ein Müllsack schützt die Bauteile vor Feuchtigkeit etc.