

Index

1. Ausstattung
2. Installation des Betriebssystems
3. Verwendung des Druckknopfes
4. Verwendung der RGB LED
5. Support

Sehr geehrter Kunde,

vielen Dank, dass Sie sich für unser Produkt entschieden haben.
 Im Folgenden haben wir aufgelistet, was bei der Inbetriebnahme zu beachten ist:

1. Ausstattung

Das Raspberry Pi 3 Model B Advanced Set besteht aus den folgenden Komponenten:

Bild	Bezeichnung	Anzahl
	<p>Raspberry Pi 3 Model B</p>	<p>5</p>
	<p>Gehäuse mit aufklappbarem Deckel</p>	<p>5</p>
	<p>Koffer</p>	<p>1</p>
	<p>Netzteil mit Steckeradaptern für Europa und Amerika</p>	<p>5</p>

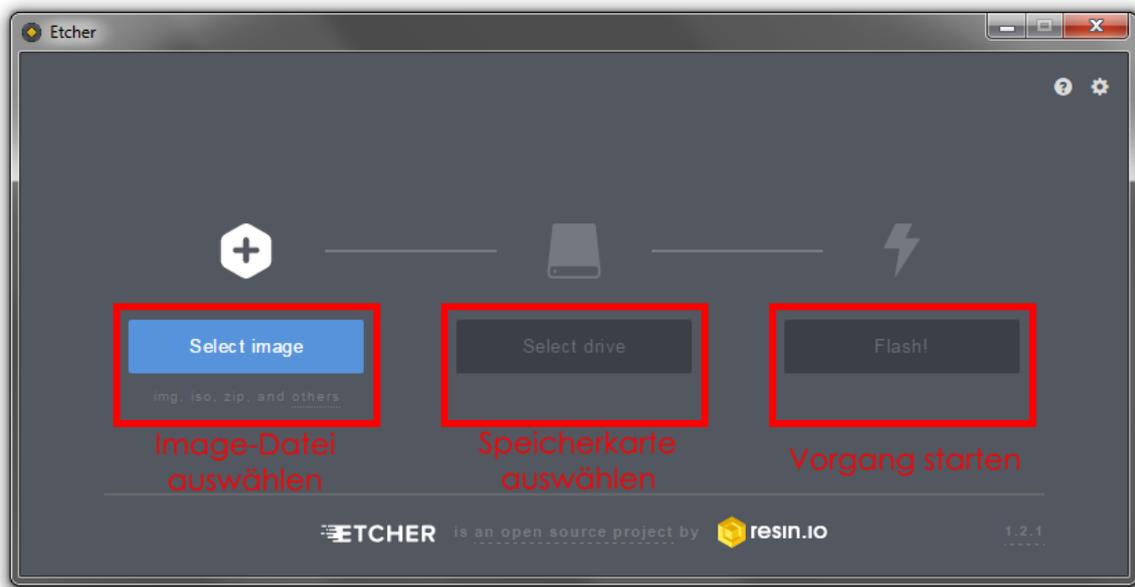
	<p>16GB Speicherkarte</p>	<p>5</p>
	<p>HDMI Kabel</p>	<p>5</p>
	<p>Breadboard</p>	<p>5</p>
	<p>Kühlkörper-Set</p>	<p>5</p>
	<p>Steckbrücken-Kabelset</p>	<p>5</p>
	<p>RGB-LED</p>	<p>5</p>
	<p>Druckknopf</p>	<p>5</p>

	<p>Lichtsensor</p>	<p>5</p>
---	--------------------	----------

2. Installation des Betriebssystems

Bevor der Raspberry Pi in Betrieb genommen werden kann muss zunächst das nötige Betriebssystem auf der beiliegenden 16GB Speicherkarte installiert werden. Schließen Sie dazu die Speicherkarte an Ihrem Computer an und laden Sie sich die aktuelle Version des [Raspbian-Betriebssystems](#) herunter.

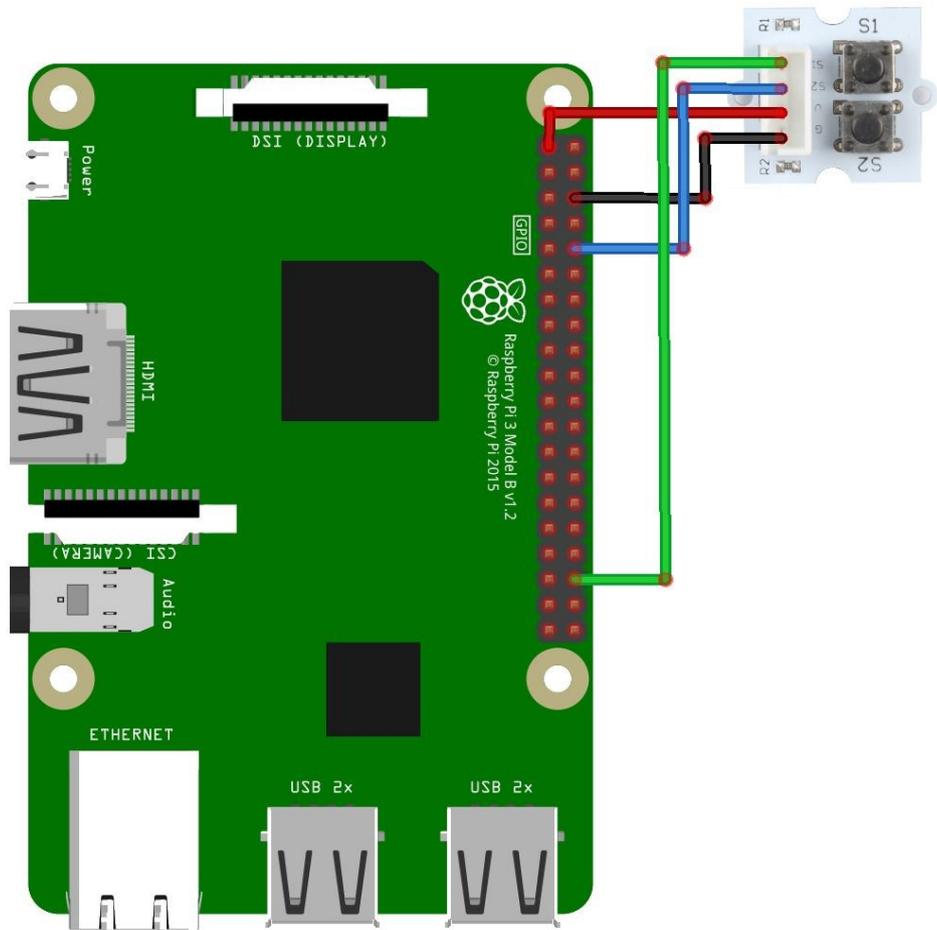
Die heruntergeladene Image-Datei können Sie mit Hilfe des Programms „[Etcher](#)“ auf Ihrer Speicherkarte installieren.



Nachdem der Vorgang abgeschlossen ist, können Sie die Speicherkarte in Ihren Raspberry Pi einlegen und ihn mit dem Netzteil und dem HDMI-Kabel in Betrieb nehmen.

3. Verwendung des Druckknopfes

Verbinden Sie die RGB-LED mit dem Raspberry Pi wie auf dem folgenden Bild, bzw. in der folgenden Tabelle, zu sehen ist.
 Verwenden Sie das beiliegende Steckbrücken-Kabelset.



Raspberry Pi Pin	Button Pin
+3,3V (Pin 1)	VCC
GND (Pin 6)	GND
GPIO15 (Pin 10)	S1
GPIO16 (Pin 36)	S2

Um nun das Druckknopf-Modul in Betrieb zu nehmen, muss eine neue Datei mit dem folgenden Befehl erstellt werden:

```
sudo nano button.py
```

Die Datei sollte den folgenden Inhalt enthalten:

```
import RPi.GPIO as GPIO

#Initialisiere Button auf Digital-PIN 15 & 16
button1 = 15
button2 = 16

GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(button1, GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(button2, GPIO.IN, pull_up_down=GPIO.PUD_UP)

while True:
    if GPIO.input(button1) == GPIO.HIGH:
        print "Button1 gedrueckt"
    if GPIO.input(button2) == GPIO.HIGH:
        print "Button2 gedrueckt"
```

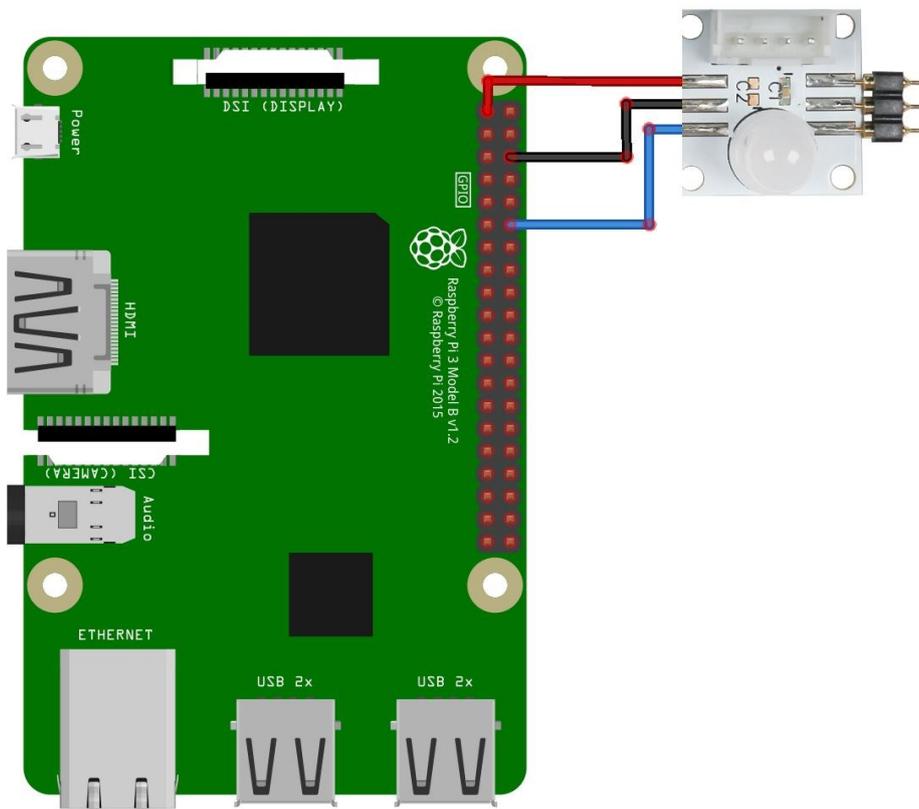
Mit der Tastenkombination **STRG+X** und einem darauffolgenden **Y** und **Enter** werden die Änderungen gespeichert und der Editor verlassen.

Das Programm kann nun mit dem folgenden Befehl ausgeführt werden:

```
sudo python button.py
```

4. Verwendung der RGB LED

Verbinden Sie die RGB-LED mit dem Raspberry Pi wie auf dem folgenden Bild, bzw. in der folgenden Tabelle, zu sehen ist.
Verwenden Sie das beiliegende Steckbrücken-Kabelset.



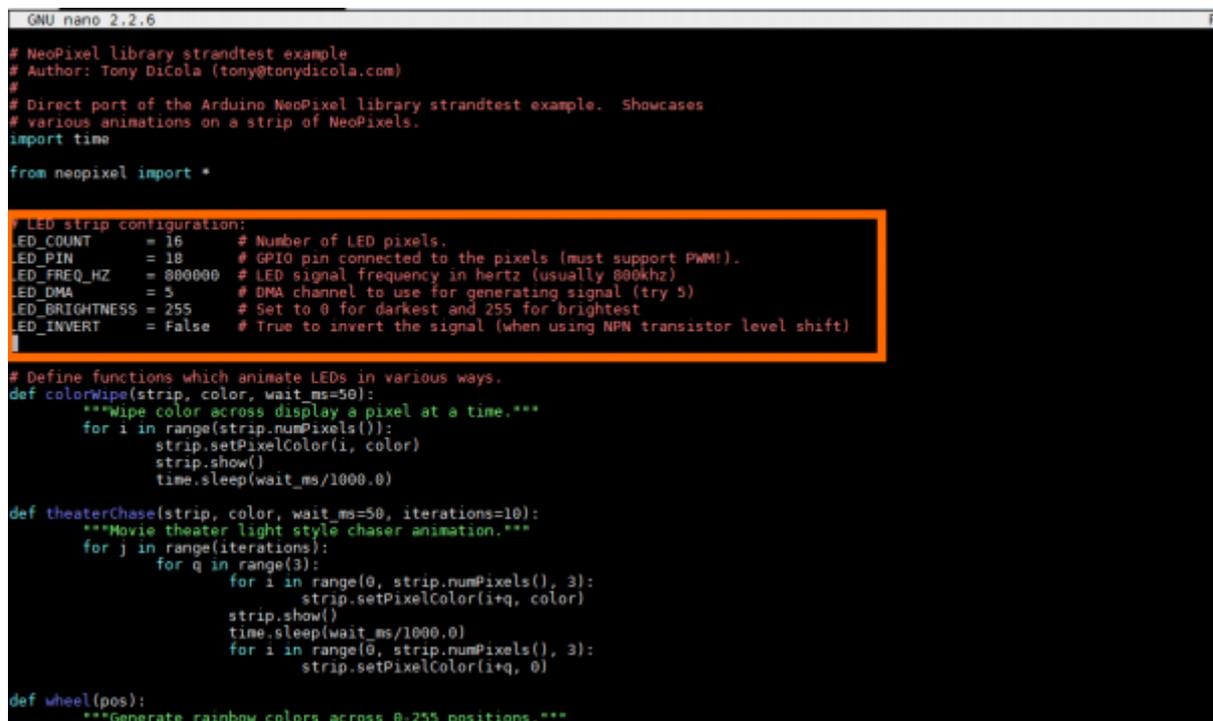
Raspberry Pi Pin	RGB LED Pin
+3,3V (Pin 1)	VCC
GND (Pin 6)	GND
GPIO18 (Pin 12)	DIN

Um die RGB-LED nun in Betrieb zu nehmen, wird zunächst die passende Bibliothek benötigt. Um diese herunterzuladen, öffnen Sie auf Ihrem Raspberry Pi das Terminal und geben Sie die folgenden Befehle ein:

```
sudo apt-get update
sudo apt-get install build-essential python-dev git scones swig
git clone https://github.com/jgarfff/rpi_ws281x
cd rpi_ws281x
sudo apt-get install scones
sudo scones
cd python
sudo python ez_setup.py
sudo python setup.py install
cd examples/
```

In dem nun geöffneten Ordner „examples“ befinden sich zwei Beispiele. Wir verwenden das Beispiel „strandtest.py“, welches jedoch zunächst angepasst werden muss. Hierzu kann die Datei mit dem folgenden Befehl geöffnet werden:

```
nano strandtest.py
```



```
GNU nano 2.2.6
# NeoPixel library strandtest example
# Author: Tony DiCola (tony@tonydicola.com)
#
# Direct port of the Arduino NeoPixel library strandtest example. Showcases
# various animations on a strip of NeoPixels.
import time

from neopixel import *

# LED strip configuration:
LED_COUNT      = 16      # Number of LED pixels.
LED_PIN        = 18      # GPIO pin connected to the pixels (must support PWM!).
LED_FREQ_HZ    = 800000  # LED signal frequency in hertz (usually 800khz)
LED_DMA        = 5       # DMA channel to use for generating signal (try 5)
LED_BRIGHTNESS = 255    # Set to 0 for darkest and 255 for brightest
LED_INVERT     = False   # True to invert the signal (when using NPN transistor level shift)

# Define functions which animate LEDs in various ways.
def colorWipe(strip, color, wait_ms=50):
    """Wipe color across display a pixel at a time."""
    for i in range(strip.numPixels()):
        strip.setPixelColor(i, color)
        strip.show()
        time.sleep(wait_ms/1000.0)

def theaterChase(strip, color, wait_ms=50, iterations=10):
    """Movie theater light style chaser animation."""
    for j in range(iterations):
        for q in range(3):
            for i in range(0, strip.numPixels(), 3):
                strip.setPixelColor(i+q, color)
            strip.show()
            time.sleep(wait_ms/1000.0)
            for i in range(0, strip.numPixels(), 3):
                strip.setPixelColor(i+q, 0)

def wheel(pos):
    """Generate rainbow colors across 0-255 positions."""
```

Die Datei besitzt einen Header, in dem verschiedene Konfigurationsparameter eingestellt werden könnten. Diese sind wie folgt erklärt:

LED_COUNT: Anzahl der angeschlossenen LEDs. Bei einer einzelnen angeschlossenen LED muss dieser Wert auf „1“ gestellt werden.

LED_PIN: Auswahl des GPIOs, an dem die Datenleitung (S1) angeschlossen ist. In unserem Fall wird GPIO18 auf Pin 12 verwendet, daher ist hier „12“ einzutragen.

LED_BRIGHTNESS: Definiert die Helligkeit der LED. Ein niedriger Wert resultiert in einer schwachen Helligkeit, ein hoher Wert in einer starken Helligkeit.

Mit der Tastenkombination **STRG+X** und einem darauffolgenden **Y** und **Enter** werden die Änderungen gespeichert und der Editor verlassen.

Das Programm kann nun mit dem folgenden Befehl ausgeführt werden:

```
sudo python strandtest.py
```

5. Support

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: service@joy-it.net

Ticket-System: <http://support.joy-it.net>

Telefon: +49 (0)2845 98469 – 66 (11- 18 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net