

FARBSENSOR

SEN-Color



1. ALLGEMEINE INFORMATIONEN

Sehr geehrte*r Kunde*in,
vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen Sie Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

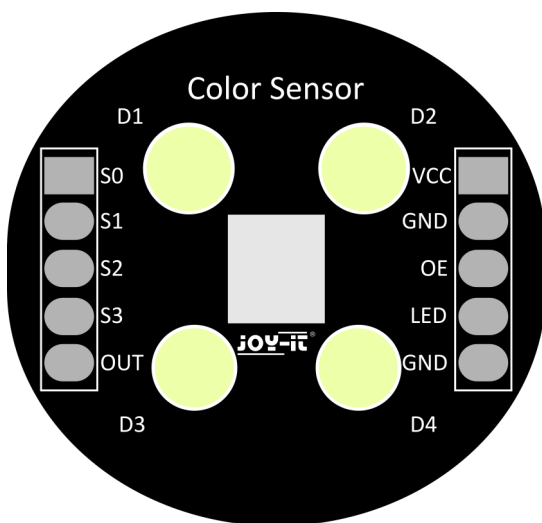
1.1 BESCHREIBUNG DES SENSORS

Der TCS3200-Chip ist so konzipiert, dass er die Farbe des auf ihn einfallenden Lichts erkennt. Er verfügt über ein Array von Fotodioden (eine Matrix von 8x8, also insgesamt 64 Sensoren). Diese Fotodioden sind mit vier Arten von Filtern bedeckt. Sechzehn Sensoren sind mit einem ROTEN Filter versehen und können somit nur den Rotanteil im einfallenden Licht messen. Weitere sechzehn haben einen GRÜNEN Filter und sechzehn einen BLAUEN Filter. Jede sichtbare Farbe kann in drei Grundfarben unterteilt werden. Diese drei Arten von gefilterten Sensoren helfen also bei der Messung der Gewichtung jeder der Primärfarben im einfallenden Licht. Die restlichen 16 Sensoren haben einen klaren Filter.

Die Ausgangswellenform ist eine Rechteckwelle mit 50 % Tastverhältnis. Sie können den Timer einer MCU verwenden, um die Periode des Pulses zu messen, um so die Frequenz zu ermitteln. Der Ausgang des TCS3200 ist einzeilig verfügbar, was bedeutet, dass nur 1 Pin als Output zur Verfügung steht.

Die Intensität der Kanäle Rot, Grün, Blau und Clear können ermittelt werden, indem die Eingänge S2 und S3, welche zur Auswahl der Sensoren verwendet werden, ihren Ausgang auf der Ausgangsleitung zur Verfügung stellen. Das heißt, dass mit den Eingängen S2 und S3 gesteuert wird, welche Art von Sensoren ihr Signal auf die Ausgangsleitung geben. Zudem gibt es noch zwei weitere Eingänge S0 und S1, welche dazu dienen, die Ausgangsfrequenz zu regulieren. Das heißt, dass mit diesen Eingängen die Intensität der Frequenz der Kanäle ROT, GRÜN, BLAU und CLEAR reguliert werden kann.

1.2 PINBELEGUNG DES SENSORS



Pin	Beschreibung
GND	Masse der Spannungsversorgung. Alle Spannungen sind auf GND referenziert.
OE	Freigabe für OUT (active low)
OUT	Ausgangsfrequenz
S0, S1	Eingänge zur Auswahl der Skalierung der Ausgangsfrequenz.
S2, S3	Eingänge zur Auswahl des Photodiodentyps
VDD	Versorgungsspannung

1.3 FARBFILTER UND FREQUENZ-SKALIERUNG

Wenn Sie einen Farbfilter für den TCS3200 auswählen, kann dieser nur diese bestimmte Farbe durchlassen, daher blockiert er andere Farben währenddessen. Die Intensität der Kanäle Rot, Grün, Blau und Clear können ermittelt werden, indem die Eingänge S2 und S3, welche zur Auswahl der Sensoren verwendet werden, ihren Ausgang auf der Ausgangsleitung zur Verfügung stellen.

Der TCS3200 hat vier Photodiodentypen. Rot, blau, grün und klar, wodurch die Amplitude des einfallenden Lichts stark vereinheitlicht wird, um die Genauigkeit zu erhöhen und die Optik zu vereinfachen, wenn das Licht auf den TCS3200 projiziert wird. Sie können den unterschiedlichen Typ der Fotodiode durch verschiedene Kombinationen von S2 und S3 wählen.

S2	S3	Photodiodentyp
Low	Low	Rot
Low	High	Blau
High	Low	Klar
High	High	Grün

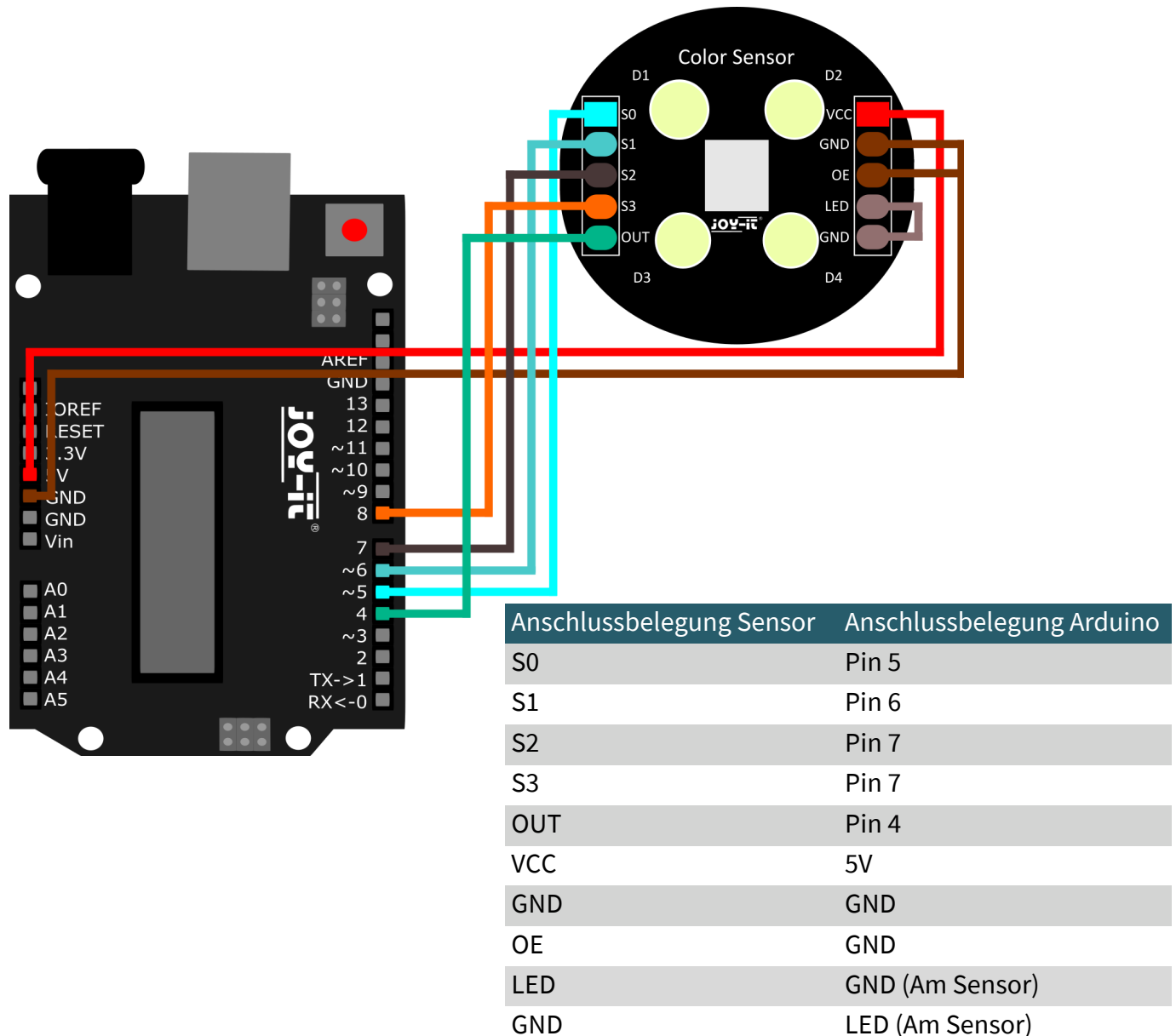
Der TCS3200 kann die Frequenz verschiedener gemessener Rechteckwellen ausgeben, welche unterschiedlichen Farben und Lichtintensitäten entsprechen. Es besteht ein Bezug zwischen dem Ausgang und der Lichtintensität, also entspricht die Ausgabe des Sensors, beeinflusst durch die Frequenzskalierung, der gemessenen Lichtintensität. Der Bereich der typischen Ausgangsfrequenz ist 2 Hz ~ 500 kHz. Wir können verschiedene Skalierungsfaktoren durch verschiedene Kombinationen von S0 und S1 erhalten. Diese Skalierung macht sich bemerkbar dadurch, dass die Größe der ausgegebenen Werte sich verändert, z.B. um die Werte an den Anforderungsbereich entsprechender Anwendungen anzupassen, so dass diese eher in den Rahmen der Anwendung passen. Zudem kann die Skalierung auch angepasst werden, um die Genauigkeit der ausgegebenen Werte zu beeinflussen.

S0	S1	Ausgabefrequenzskalierung
Low	Low	AUS
Low	High	2%
High	Low	20%
High	High	100%

2. VERWENDUNG MIT DEM ARDUINO

Im Folgenden ist nun zu sehen, wie Sie Ihren Farbsensor an Ihren Arduino anzuschließen haben. Hierbei folgen Sie bitte der Anschlussbelegungstabelle und der Abbildung.

Bitte beachten Sie, dass der Sensor vor jeder Verwendung kalibriert werden sollte. Weitere Informationen zur Kalibrierung finden Sie in Abschnitt 4.1.



2.1 CODEBEISPIEL ARDUINO

Nach dem Anschließen des SEN-Color, wie oben gezeigt, können Sie nun den Arduino per beiliegendem Kabel an Ihren PC anschließen. Danach können Sie Ihre Arduino IDE öffnen. Sollten Sie diese noch nicht installiert haben, dann können Sie diese [hier](#) herunterladen und installieren. Danach können Sie unter **Werkzeuge > Port** Ihren Arduino auswählen. Unter **Werkzeuge > Board** wählen Sie "Arduino Uno" und unter **Werkzeuge > Programmer** wählen Sie "ArduinoISP".

Nach den Oben genannten Schritten können Sie nun folgenden Code einfach in Ihre IDE kopieren. Wenn Sie nun diesen Code auf Ihren Arduino geladen haben, können Sie den seriellen Monitor, welchen Sie in der oberen rechten Ecke Ihrer IDE finden, öffnen. Den seriellen Monitor müssen Sie auf 9600 Baud einstellen, wenn dieser nicht bereits auf 9600 Baud eingestellt ist. Sollte all dies gelungen sein und der Sensor ordnungsgemäß kalibriert sein, werden Ihnen nun Ihre Messwerte angezeigt im Bereich von 0-255.

```

#include <math.h>
#define noPaverageseInterrupts 1

// Zuordnung der Sensor Pins
#define S0 5
#define S1 6
#define S2 7
#define S3 8
#define sensorOut 4

/*Kalibrierungs Werte (müssen vor
  jeder benutzung aktualisiert werden)*/
int redMin = 0;
int redMax = 1;
int greenMin = 0;
int greenMax = 1;
int blueMin = 0;
int blueMax = 1;

int redColor = 0;
int greenColor = 0;
int blueColor = 0;

int redFrequency = 0;
int redEdgeTime = 0;
int greenFrequency = 0;
int greenEdgeTime = 0;
int blueFrequency = 0;
int blueEdgeTime = 0;

void setup()
{
  /*Definierung der Sensor Pins*/
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(sensorOut, INPUT);

  /*Skalierung der Ausgabe Frequenz
    S0/S1
    LOW/LOW=AUS, LOW/HIGH=2%,
    HIGH/LOW=20%, HIGH/HIGH=100%*/
  digitalWrite(S0, HIGH);
  digitalWrite(S1, HIGH);

  Serial.begin(9600);
}

//Initialisierung der Schleife
void loop() {

  Serial.println("-----");

  {
    /*Bestimmung des Photodioden Typs bei der Messung
      S2/S3
      LOW/LOW=RED, LOW/HIGH=BLUE,
      HIGH/HIGH=GREEN, HIGH/LOW=CLEAR*/
    digitalWrite(S2, LOW);
    digitalWrite(S3, LOW);
  }
}

```

```

    /*Frequenzmessung der bestimmten Farbe und dessen Zuord-
nung zu einem RGB-Wert zwischen 0-255*/
    float(redEdgeTime) = pulseIn(sensorOut, HIGH) + pulseIn
(sensorOut, LOW);
    float(redFrequency) = (1 / (redEdgeTime / 1000000));
    redColor = map(redFrequency, redMax, redMin, 255, 0);
    if (redColor > 255) {
        redColor = 255;
    }
    if (redColor < 0) {
        redColor = 0;
    }
    /*Ausgabe der Frequenz gemappt zu 0-255*/
    Serial.print("Red Frequency: ");
    Serial.println(redFrequency);
    Serial.print("R = ");
    Serial.println(redColor);
    delay(100);
}

{
    digitalWrite(S2, HIGH);
    digitalWrite(S3, HIGH);
    /*Frequenzmessung der bestimmten Farbe und dessen Zuord-
nung zu einem RGB-Wert zwischen 0-255*/
    float(greenEdgeTime) = pulseIn(sensorOut, HIGH) + pulseIn
(sensorOut, LOW);
    float(greenFrequency) = (1 / (greenEdgeTime / 1000000));
    greenColor = map(greenFrequency, greenMax, greenMin, 255,
0);
    if (greenColor > 255) {
        greenColor = 255;
    }
    if (greenColor < 0) {
        greenColor = 0;
    }
    /*Ausgabe der Frequenz gemappt zu 0-255*/
    Serial.print("Green Frequency: ");
    Serial.println(greenFrequency);
    Serial.print("G = ");
    Serial.println(greenColor);
    delay(100);
}

{
    digitalWrite(S2, LOW);
    digitalWrite(S3, HIGH);
    /*Frequenzmessung der bestimmten Farbe und dessen Zuord-
nung zu einem RGB-Wert zwischen 0-255*/
    float(blueEdgeTime) = pulseIn(sensorOut, HIGH) + pulseIn
(sensorOut, LOW);
    float(blueFrequency) = (1 / (blueEdgeTime / 1000000));
    blueColor = map(blueFrequency, blueMax, blueMin, 255, 0);
    if (blueColor > 255) {
        blueColor = 255;
    }
    if (blueColor < 0) {
        blueColor = 0;
    }
    /*Ausgabe der Frequenz gemappt zu 0-255*/
    Serial.print("Blue Frequency: ");
    Serial.println(blueFrequency);
    Serial.print("B = ");
    Serial.println(blueColor);
}

```

```
    delay(100);
}

Serial.println("-----");
delay(100);

/*Bestimmung der gemessenen Farbe durch Abgleich
mit den Werten der Anderen gemessenen Farben*/
if (redColor > greenColor and redColor > blueColor) {
    Serial.println("Red detected ");
    delay(100);
}

if (greenColor > redColor and greenColor > blueColor) {
    Serial.println("Green detected ");
    delay(100);
}

if (blueColor > redColor and blueColor > greenColor) {
    Serial.println("Blue detected ");
    delay(100);
}

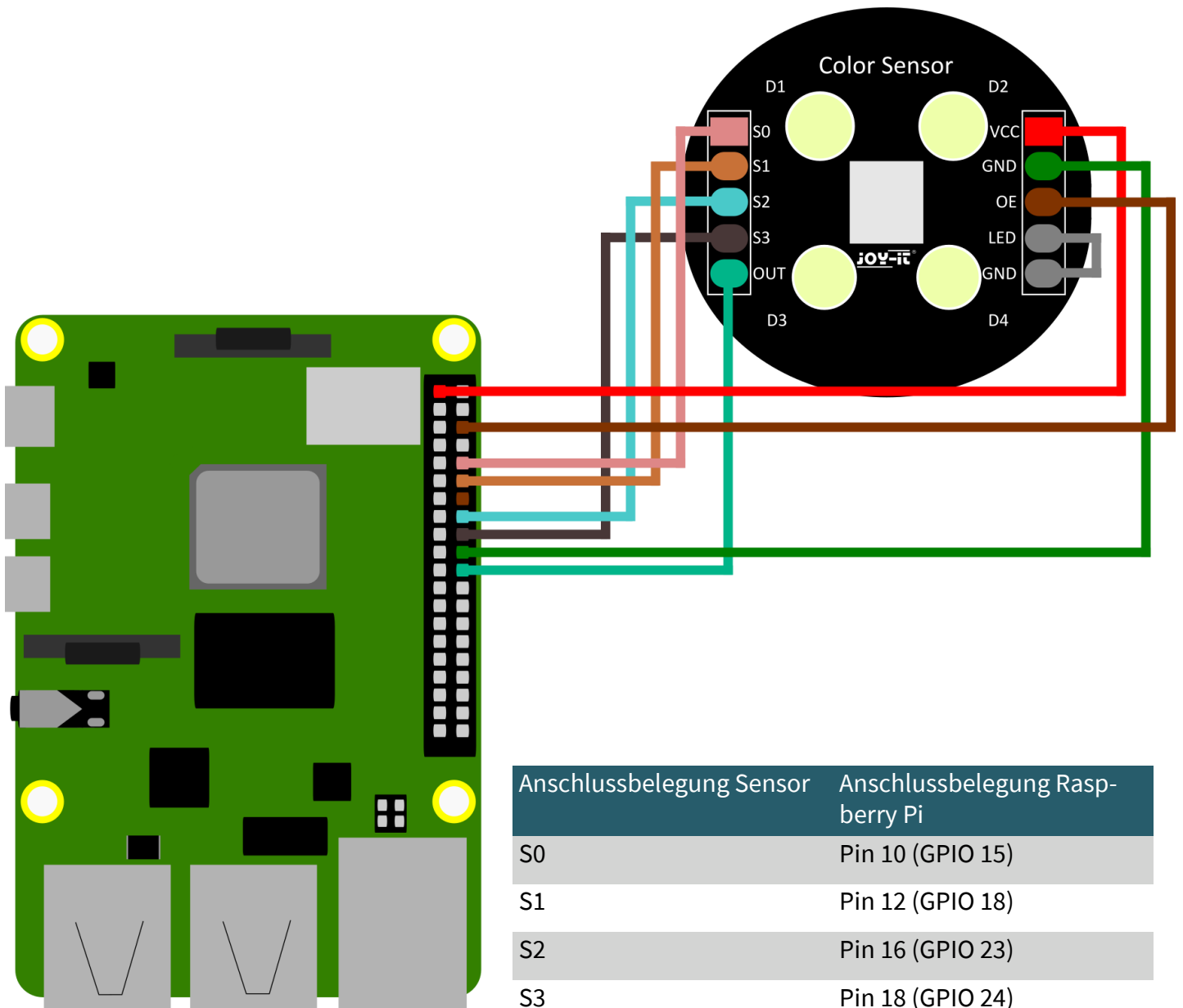
if (blueColor > 0 and greenColor > 0 and redColor > 0) {
    Serial.println("Please place an Object in front of the
Sensor");
    delay(100);
}

Serial.println("-----");
delay(1000);
}
```

3. VERWENDUNG MIT DEM RASPBERRY PI

Im Folgenden ist nun zu sehen, wie Sie Ihren Farbsensor an Ihren Raspberry Pi anzuschließen haben. Hierbei folgen Sie bitte der Anschlussbelegungstabelle und der Abbildung. (Zum Erzielen genauerer Messergebnisse empfehlen wir Ihnen einen Voltage Translator zu benutzen, wie unser [KY-051](#), um dann den Sensor mit 5 V zu betreiben.)

Bitte beachten Sie, dass der Sensor vor jeder Verwendung kalibriert werden sollte. Weitere Informationen zur Kalibrierung finden Sie in Abschnitt 4.1.



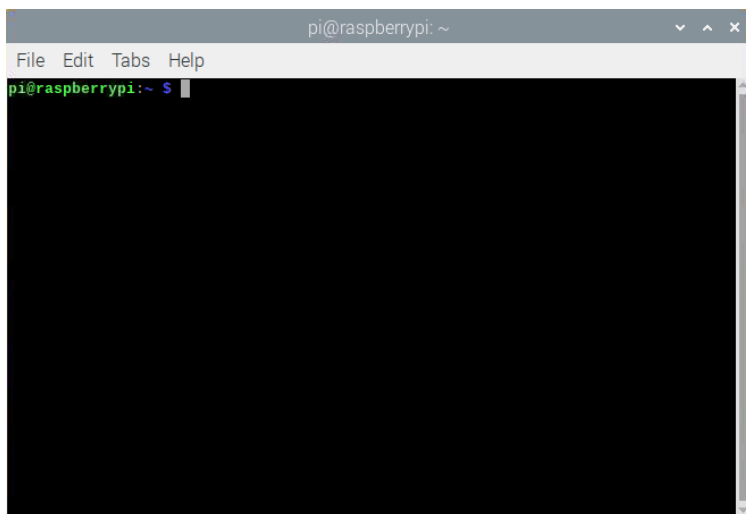
Anschlussbelegung Sensor	Anschlussbelegung Rasp- berry Pi
S0	Pin 10 (GPIO 15)
S1	Pin 12 (GPIO 18)
S2	Pin 16 (GPIO 23)
S3	Pin 18 (GPIO 24)
VCC	Pin 1 / 3V3
GND	Pin 20 / GND
OE	Pin 6 / GND
LED	GND (Am Sensor)
GND	LED (Am Sensor)
OUT	Pin 22 (GPIO 25)

3.1 CODEBEIPIEL RASPBERRY PI

Nach dem Anschließen des SEN-Color, wie zuvor gezeigt, können Sie nun den Raspberry Pi an einen Bildschirm sowie eine Tastatur und eine Maus anschließen. Nach dem erstmaligen Einrichten des Raspberry Pi können Sie nun die Konsole öffnen.



Nach den oben genannten Schritten sollten Sie folgendes Fenster vor Ihnen sehen.



Als nächsten Schritt werden Sie nun eine neue Datei erstellen, indem Sie

```
sudo nano SEN-Color.py
```

eingeben und in die dann geöffnete neue Datei den folgenden Code hereinkopieren, indem Sie auf **Bearbeiten > Einfügen** gehen.

Bitte beachten Sie, dass die Kalibrierungswerte aktualisiert werden müssen, bevor Sie den Sensor einwandfrei benutzen können!

```
import RPi.GPIO as GPIO
import time

#Zuordnung der Sensor Pins
s0 = 15
s1 = 18
s2 = 23
s3 = 24
signal = 25

NUM_CYCLES = 500

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

def setup():
```

```

GPIO.setup(signal,GPIO.IN, pull_up_down=GPIO.PUD_UP)
GPIO.setup(s0,GPIO.OUT)
GPIO.setup(s1,GPIO.OUT)
GPIO.setup(s2,GPIO.OUT)
GPIO.setup(s3,GPIO.OUT)

def calc(raw, cN, out):
    #Kalibrierungs Werte (muessen vor
    #jeder benutzung aktualisiert werden)
    #SIEHE ANLEITUNG
    redMin = 0
    redMax = 1
    greenMin = 0
    greenMax = 1
    blueMin = 0
    blueMax = 1

    out_from = 255
    out_to = 0

    colourTime = 1
    #Frequenzmessung der bestimmten Farbe
    for impulse_count in range(NUM_CYCLES):
        GPIO.wait_for_edge(signal, GPIO.RISING)
        if colourTime is 1:
            start = time.time()
            colourTime = 0
        duration = time.time() - start
        raw = (NUM_CYCLES / duration)
        print("red frequency - ",raw,"Hz")

    #zuordnung der gemessenen Farbe zu
    #einem RGB-Wert zwischen 0-255
    out_range = out_from - out_to
    in_range = redMax - redMin
    in_val = raw - redMin
    val = (float(in_val) / in_range) * out_range
    out = out_to + val
    if out < 0:
        out = 0
    elif out > 255:
        out = 255
    print(cN,"= ",out)

def loop():
    temp = 1

    #Bestimmung der Output-Frequenz Skalierung
    #S0/S1
    #LOW/LOW=/, LOW/HIGH=2%,
    #HIGH/LOW=20%, HIGH/HIGH=100%
    GPIO.output(s0,GPIO.HIGH)
    GPIO.output(s1,GPIO.HIGH)

    while(1):

#####
#####
        time.sleep(0.3)
        print("-----")
        time.sleep(0.3)

```

```

print("Measured frequencys of Red, Green and Blue in Hz")

#Bestimmung des Photodioden Typs
#S2/S3
#LOW/LOW=RED, LOW/HIGH=BLUE,
#HIGH/HIGH=GREEN, HIGH/LOW=CLEAR

GPIO.output(s2,GPIO.LOW)
GPIO.output(s3,GPIO.LOW)
time.sleep(0.3)
redraw = 0
out_val1 = 0
calc(redraw, "R", out_val1)

#####
#####

GPIO.output(s2,GPIO.HIGH)
GPIO.output(s3,GPIO.HIGH)
time.sleep(0.3)
greenraw = 0
out_val2 = 0
calc(greenraw, "G", out_val2)

#####
#####

GPIO.output(s2,GPIO.LOW)
GPIO.output(s3,GPIO.HIGH)
time.sleep(0.3)
blueraw = 0
out_val3 = 0
calc(blueraw, "B", out_val3)

#####
#####

red = out_val1
green = out_val2
blue = out_val3

time.sleep(0.3)
print("-----")
print("-----")
time.sleep(0.3)

#Bestimmung der gemessenen Farbe durch abgleich
#mit den Werten der anderen gemessenen Farben

if red>green and blue:
    print("Red detected")
    temp=1
elif green>red and blue:
    print("Green detected")
    temp=1
elif blue>red and green:
    print("Blue detected")
    temp=1

```

```

elif blue>red and green>red:
    print("Turquoise detected")
    temp=1
elif blue>green and red>green:
    print("Purple detected")
    temp=1
elif temp==1:
    print("place a object infront of the Sensor")
    temp=0

time.sleep(0.3)

def endprogram():
    GPIO.cleanup()

if __name__=='__main__':

    setup()

    try:
        loop()

    except KeyboardInterrupt:
        endprogram

```

Nach dem Kopieren und Einfügen des Codes speichern und verlassen Sie die Datei mit **STR+O ENTER STR+X**. Den Code führen Sie dann mit der Eingabe von

```
sudo python3 SEN-Color.py
```

aus. Den Code beenden Sie letztlich mit der Eingabe von **STR+C**.

4. KALIBRIERUNG DES SENSORS

Grundsätzlich ist bei der Kalibrierung zu beachten, dass der Sensor bereits in den Lichtverhältnissen kalibriert wird, in denen er später arbeiten soll. Es empfiehlt sich daher, den Sensor bereits vor der Kalibrierung in seiner endgültigen Arbeitsposition zu montieren. Da der Sensor zudem empfindlich auf Streulicht reagieren kann, kann ggf. ein zusätzlicher Streulichtschutz die Messergebnisse verbessern.

4.1 KALIBRIERUNG DES SENSORS AM ARDUINO/RASPBERRY PI

Für eine möglichst genaue Kalibrierung sollte der Farbsensor während der Kalibrierungsmessung einen Abstand von ca. 4-5 cm zum Messobjekt haben. Zusätzlich empfiehlt sich die Kalibrierung ausschließlich mit den Grundfarben (Rot, Grün und Blau) durchzuführen. Hierfür können Sie sich unser [RGB-Kalibrierungsblatt](#) herunterladen und ausdrucken.

Bei der Kalibrierung werden für jede Grundfarbe die minimalen und maximalen Werte ermittelt und in den Code eingetragen.

```
#define S1 6
#define S2 7
#define S3 8
#define sensorOut 4

/*Kalibrierungs Werte (müssen vor
jeder Benutzung aktualisiert werden)*/
int redMin = 0;
int redMax = 1;
int greenMin = 0;
int greenMax = 1;
int blueMin = 0;
int blueMax = 1;
```

*Arduino Codeausschnitt

```
def loop():
    temp = 1
    #Kalibrierungs Werte (muessen vor
    #jeder Benutzung aktualisiert werden)
    #SIEHE ANLEITUNG
    redMin = 0
    redMax = 1
    greenMin = 0
    greenMax = 1
    blueMin = 0
    blueMax = 1
```

*Raspberry Pi Codeausschnitt

Führen Sie nun den Code aus 2.1 oder 3.1 aus. Zur Ermittlung der minimalen Messwerte werden Messungen ohne Messobjekt ausgeführt. Übernehmen Sie nun die Messwerte der einzelnen Kanäle für die Minimalen Kalibrierungswerte.

Tipp: Um eine noch bessere Kalibrierung zu erzielen können Sie auch den Durchschnitt von mehreren Messungen verwenden.

Zur Ermittlung der Maximal-Kalibrierungswerte werden nun Messungen mit den Grundfarben als Messobjekt durchgeführt. Hier empfiehlt sich unser [RGB-Kalibrierungsblatt](#).

Halten Sie den Sensor nacheinander vor die drei Grundfarben und übernehmen Sie die hier gemessenen Werte für die Maximalen Kalibrierungswerte.

Tipp: Auch hier können Sie die Kalibrierung verbessern, indem Sie den Durchschnitt von mehreren Messungen verwenden.

Nachdem die minimalen und maximalen Kalibrierungswerte in den ursprünglichen Code eingetragen wurden, kann der Code erneut übertragen werden. Die Kalibrierung ist damit abgeschlossen und der Sensor bereit zur Verwendung.

5. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektroggesetz (ElektroG)



Symbol auf Elektro- und Elektronikgeräten: 

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

Rückgabemöglichkeiten:

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in haushaltsüblichen Mengen abgeben werden.

Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

Möglichkeit Rückgabe in Ihrer Nähe:

Sie senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an Service@joy-it.net oder per Telefon an uns.

Informationen zur Verpackung:

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, Sie lassen Ihnen dann eine geeignete Verpackung zukommen.

6. SUPPORT

Sie sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen Sie 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 (10 - 17 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net