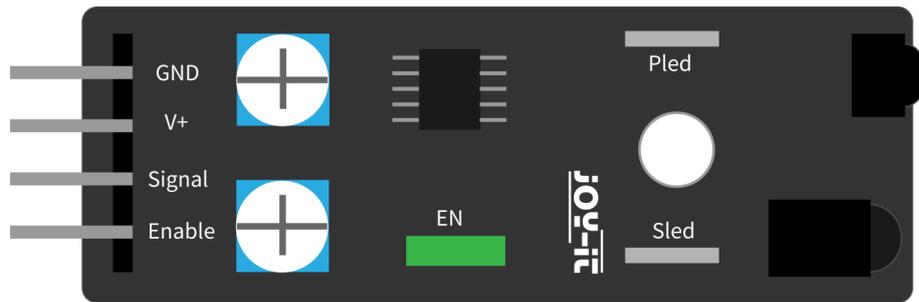


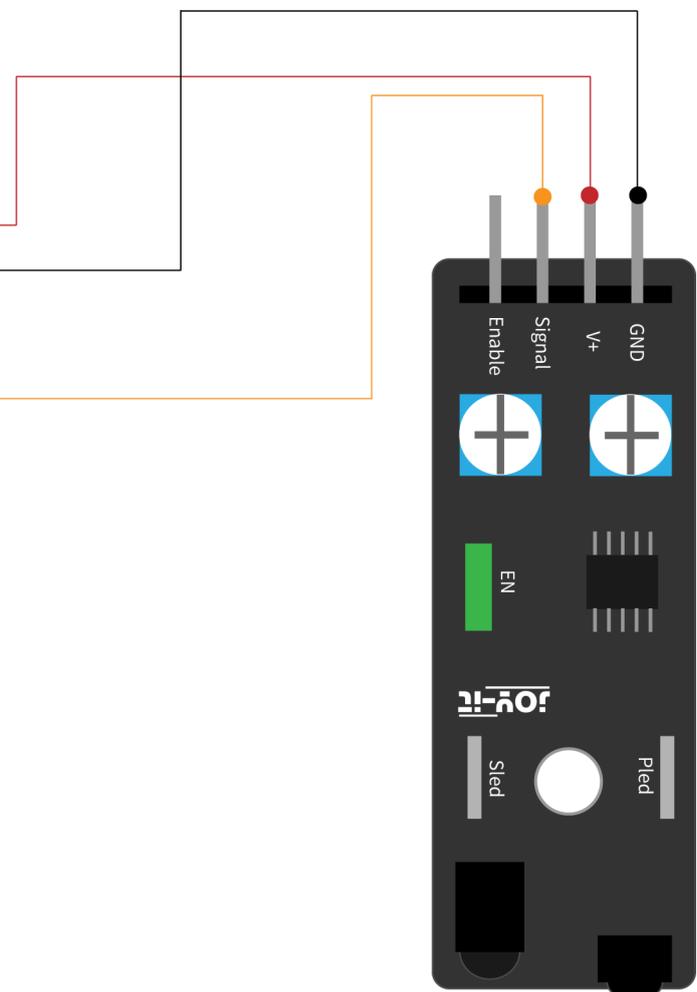
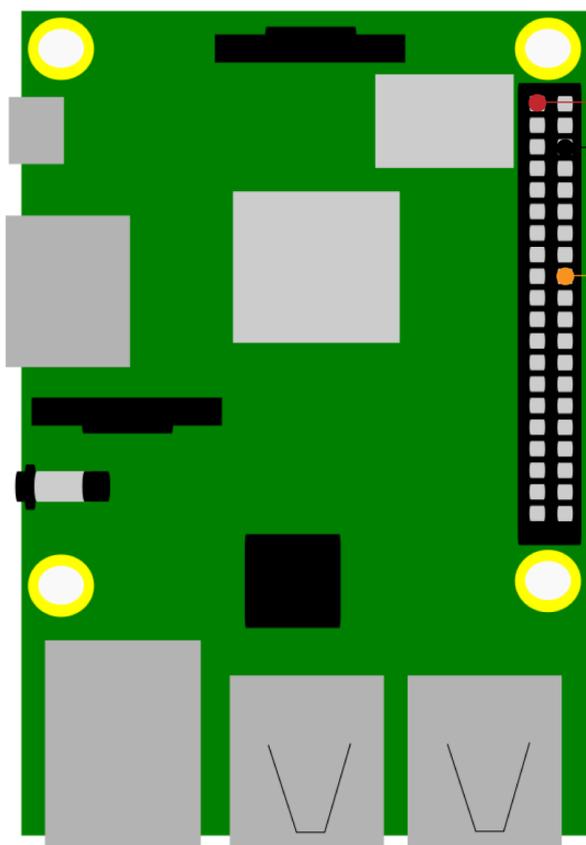
2. PINBELEGUNG



3. VERWENDUNG MIT DEM RASPBERRY PI

1. Anschluss

KY-032	Raspberry Pi
GND	Masse (Pin 6)
V+	3.3V (Pin 1)
Signal	GPIO24 (Pin 18)
Enable Pin	- Steckbrücke gesteckt



2. Beispielcode

```
# Benötigte Module werden importiert und eingerichtet
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)

# Hier wird der Eingangs-Pin deklariert, an dem der Sensor angeschlossen ist.
GPIO_PIN = 24
GPIO.setup(GPIO_PIN, GPIO.IN, pull_up_down = GPIO.PUD_UP)

# Pause zwischen der Ausgabe des Ergebnisses wird definiert (in Sekunden)
delayTime = 0.5
print ("Sensor-Test [druecken Sie STRG+C, um den Test zu beenden]")

# Hauptprogrammschleife
try:
    while True:
        if GPIO.input(GPIO_PIN) == True:
            print ("Kein Hindernis")
        else:
            print ("Hindernis erkannt")
        print ("-----")

        # Reset + Delay
        time.sleep(delayTime)

# Aufräumarbeiten nachdem das Programm beendet wurde
except KeyboardInterrupt:
    GPIO.cleanup()
```

Dieser Code liest den aktuellen Status des Sensors aus und gibt in der Konsole aus, ob ein Hindernis vorliegt oder nicht. Wenn der Sensor ein Hindernis erkennt leuchtet eine weitere LED (Sled) auf. Der Sensor besitzt zwei Potentiometer mit welchen man die Empfindlichkeit des Receivers als auch des Senders einstellen kann. Sie können sich auch diesen Beispielcode [hier](#) herunterladen. Zunächst müssen Sie dieses Package installieren:

```
sudo apt install python3-rpi.gpio
```

Nun können Sie entweder mit dem folgenden Befehlen die Datei herunterladen und entpacken:

```
wget http://sensorkit.joy-it.net/images/d/d8/KY-032_RPi_HindernisDetektor.zip
unzip KY-032_RPi_HindernisDetektor.zip
```

Oder Sie erstellen eine neue Datei mit dem folgenden Befehl und kopieren den hier beschriebenen Beispielcode hinein:

```
sudo nano KY-032_RPi_HindernisDetektor.py
```

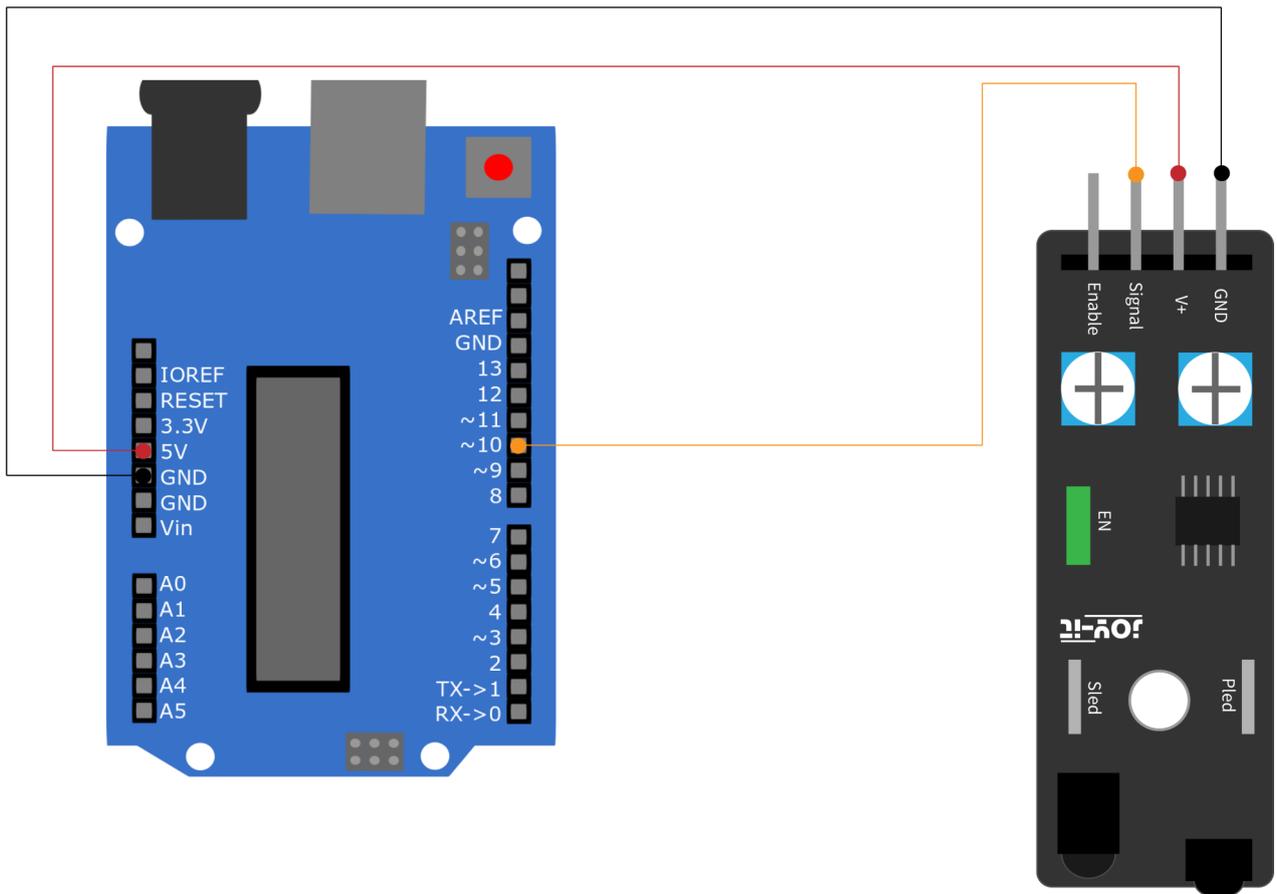
Nun müssen Sie die Datei mit **STRG + O** speichern und mit **Enter** bestätigen. Mit **STRG + X** verlassen Sie den Editor wieder.

Mit beiden Varianten wird das Programm wie folgt aufgerufen:

```
sudo python3 KY-032_RPi_HindernisDetektor.py
```

4. VERWENDUNG MIT DEM ARDUINO

1. Anschluss



KY-032	Arduino
GND	GND
V+	Pin 5V
Signal	Pin 10
Enable Pin	- Steckbrücke gesteckt

2. Codebeispiel

Dieser Code liest den aktuellen Status des Sensors aus und gibt in der Konsole aus, ob ein Hindernis vorliegt oder nicht. Wenn der Sensor ein Hindernis erkennt leuchtet eine weitere LED (Sled) auf.

```
int Sensor = 10; // Deklaration des Sensor-Eingangspin

void setup (){
  Serial.begin(9600); // Initialisierung serielle Ausgabe
  pinMode (Sensor, INPUT) ; // Initialisierung Sensorpin
}

// Das Programm liest den aktuellen Status des Sensor-Pins aus und
// gibt in der seriellen Konsole aus, ob ein Hindernis aktuell erkannt wird
// oder ob kein Hindernis sich vor dem Sensor befindet
void loop (){
  // Das gegenwärtige Signal am Sensor wird ausgelesen
  bool val = digitalRead (Sensor) ;

  // Falls ein Signal erkannt werden konnte, wird die LED eingeschaltet.
  if (val == HIGH){
    Serial.println("Kein Hindernis");
  }
  else {
    Serial.println("Hindernis erkannt");
  }
  Serial.println("-----");
  delay(500); // Pause zwischen der Messung von 500ms
}
```

In diesem Codebeispiel werden die empfangenen Daten im seriellen Monitor ausgegeben. Achten Sie dabei darauf, dass die Baudrate richtig eingestellt ist (9600).

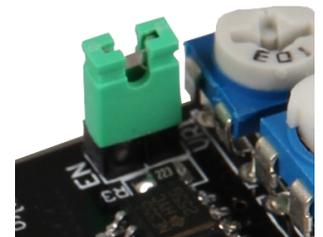
Gehen Sie auch vor dem Hochladen sicher, dass Sie das richtige Board und den richtigen Port in Ihrer Arduino IDE eingestellt haben.

Sie können sich auch das Codebeispiel [hier](#) herunterladen.

Der Sensor besitzt zwei Potentiometer mit welchen man die Empfindlichkeit des Receivers als auch des Senders einstellen kann.

5. ENABLE - PIN

Dieser Sensor besitzt einen zusätzlichen Pin namens Enable. Dieser bietet die Möglichkeit en Sensor mittels Controller zu aktivieren oder zu deaktivieren. Standardmäßig ist dies aktiviert mittels einer Steckbrücke. Dies bedeutet, dass die Erkennung von Hindernissen immer durchgeführt wird. Wenn man die dauerhafte Erkennung deaktivieren möchte, entfernt man diese Steckbrücke (EN) und verbindet den Enable-Pin mit einem Pin, mit welchem ein Steuersignal gesendet werden kann.



6. 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:

Wir 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, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

7. 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 (10 - 17 Uhr)

Für weitere Informationen besuchen Sie unsere Website:

www.joy-it.net