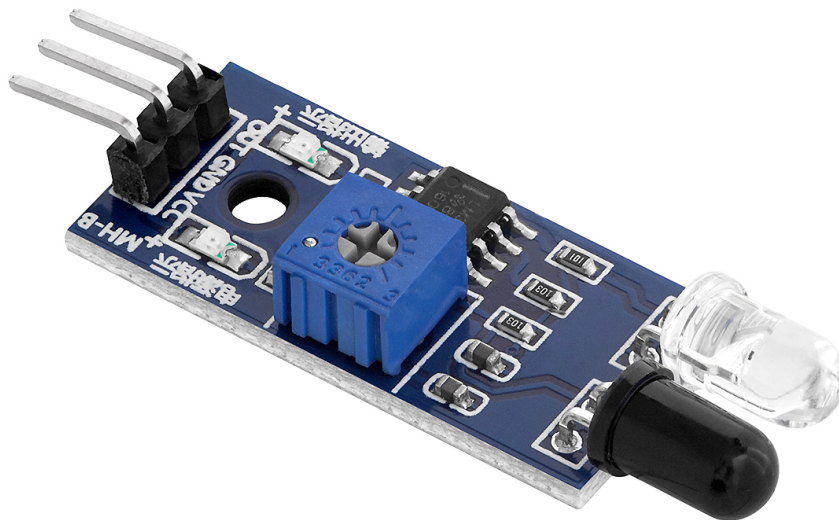


# AZ-Delivery

## Willkommen!

Vielen Dank, dass sie sich für unser *IR-Abstand-Sensor-Modul* von *AZ-Delivery* entschieden haben. In den nachfolgenden Seiten werden wir Ihnen erklären wie Sie das Gerät einrichten und nutzen können.

**Viel Spaß!**



# Az-Delivery

## Inhaltsverzeichnis

Einführung.....	3
Technische Daten.....	4
Pinbelegung.....	5
Wie man die Arduino IDE einrichtet.....	6
Wie man den Raspberry Pi und Python einrichtet.....	10
Verbindung des Moduls mit dem Uno.....	11
Sketch-Beispiel.....	12
Verbindung des Moduls mit dem Raspberry Pi.....	14
Libraries und Tools für Python.....	15
Python-Skript.....	16

# Az-Delivery

## Einführung

Das Sensormodul zur Hinderniserkennung ist ein Gerät, das zur Erkennung von Hindernissen verwendet wird. Es ist ein Mehrzweckgerät und kann in vielen Anwendungen zur Erkennung von Objekten, zur Kollisionserkennung, zur Kantenvermeidung, zur Farberkennung, zur Branderkennung, zur Linienenerkennung und als Encoder zum Zählen von Objekten eingesetzt werden. Das Modul kann Teil anderer Geräte wie Roboter, Maschinen, Sicherheitssysteme usw. sein.

Der Hinderniserkennungssensor ist ein Gerät, das Infrarotlicht aussendet und erkennt, um die in der Umgebung auftretenden Veränderungen zu erfassen. Er erkennt Licht in den Wellenlängen zwischen 0,75 und 1000µm.

Das Modul besteht aus einer LED (Light Emitting Diode) als Sender, einer Photodiode als Empfänger und einem Operationsverstärker, der als Spannungskomparator fungiert. Er misst die Spannungsschwelle der Photodiode in Reihe mit einem Widerstand. Wenn die Spannung unter den Schwellenwert fällt, ist der Schwellen-Ausgang des Operationsverstärkers HIGH und die Signal-LED leuchtet auf, was anzeigt, dass das Objekt erfasst wird.

Das Ausgangssignal ist digital und der Ausgangspin kann sich im HIGH- oder LOW-Zustand befinden. Wenn sich der Ausgangspin im HIGH-Zustand befindet, ist die Spannung maximal und wenn er sich im LOW-Zustand befindet, liegt die Spannung bei 0V.

# Az-Delivery

## Technische Daten

Betriebsspannung	3,3V bis 5V
Betriebsstrom	20mA
Erfassungsbereich	20mm bis 200mm
Erfassungswinkel	35° Grad
Ausgangssignal	Digital
Durchmesser Montageloch	3mm
Dimensionen	48x14x7mm (1.9x0.5x0.3in)

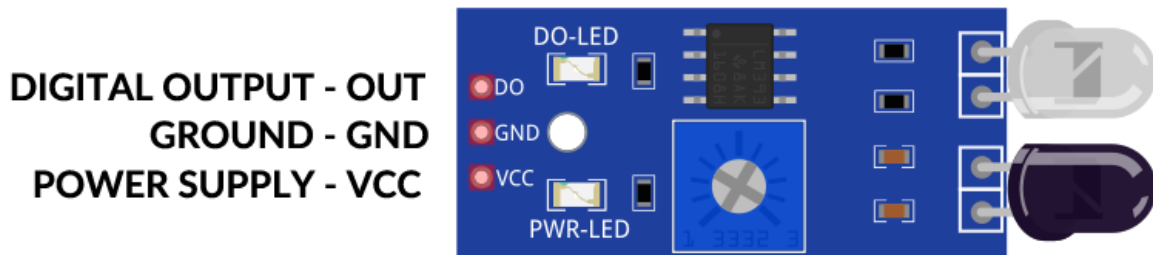
Das Modul hat eine integrierte LED, für die Leistungs- und Detektionsanzeige.

Die Empfindlichkeit des Moduls kann mit einem On-Board-Potentiometer eingestellt werden. Durch Drehen der Potentiometerwelle im Uhrzeigersinn wird die Empfindlichkeit erhöht. Durch Bewegen der Potentiometerwelle gegen den Uhrzeigersinn wird die Empfindlichkeit des Moduls verringert.

# Az-Delivery

## Pinbelegung

Das Modul hat drei Pins. Die Pinbelegung ist wie folgt:



Das Modul arbeitet mit einem Spannungsbereich von 3,3V und 5V .

**Hinweis:** Wenn Sie das Modul mit einem Raspberry Pi verwenden, verbinden Sie den VCC-Pin des Moduls mit 3,3V. Eine Verbindung dieses Pins mit 5V könnte den Raspberry Pi beschädigen.

# Az-Delivery

## Wie man die Arduino IDE einrichtet

Falls die Arduino-IDE nicht installiert ist, folgen Sie dem [link](#) und laden Sie die Installationsdatei für das Betriebssystem Ihrer Wahl herunter.

### Download the Arduino IDE



The screenshot shows the Arduino IDE download page. On the left, there is a teal circular logo with a white infinity symbol containing a minus and a plus sign. To the right of the logo, the text reads: **ARDUINO 1.8.9**. Below this, it states: "The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software. This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions." On the right side of the page, there is a teal sidebar with the following options: **Windows** Installer, for Windows XP and up; **Windows** ZIP file for non admin install; **Windows app** Requires Win 8.1 or 10 with a "Get" button; **Mac OS X** 10.8 Mountain Lion or newer; **Linux** 32 bits; **Linux** 64 bits; **Linux** ARM 32 bits; **Linux** ARM 64 bits; and links for Release Notes, Source Code, and Checksums (sha512).

Für Windows Benutzer: Doppelklicken Sie auf die heruntergeladene .exe l-Datei und folgen Sie den Anweisungen im Installationsfenster.

# Az-Delivery

Für *Linux* Benutzer, laden Sie eine Datei mit der Erweiterung *.tar.xz* herunter, die extrahiert werden muss. Wenn sie extrahiert ist, gehen Sie in das extrahierte Verzeichnis und öffnen Sie das Terminal in diesem Verzeichnis. Zwei *.sh* Skripte müssen ausgeführt werden, das erste namens *arduino-linux-setup.sh* und das zweite heißt *install.sh*.

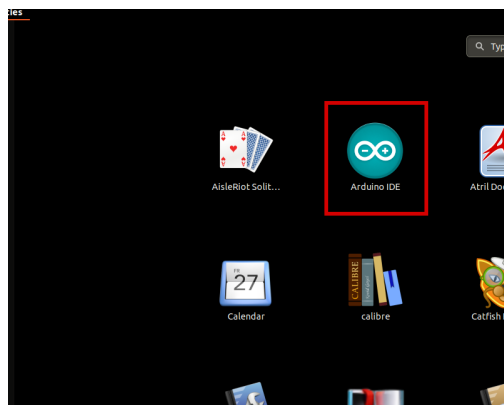
Um das erste Skript im Terminal auszuführen, öffnen Sie das Terminal im extrahierten Ordner und führen Sie den folgenden Befehl aus:

```
sh arduino-linux-setup.sh user_name
```

**user\_name** - ist der Name eines Superusers im Linux-Betriebssystem. Ein Passwort für den Superuser muss beim Start des Befehls eingegeben werden. Warten Sie einige Minuten, bis das Skript vollständig abgeschlossen ist.

Das zweite Skript mit der Bezeichnung *install.sh*-Skript muss nach der Installation des ersten Skripts verwendet werden. Führen Sie den folgenden Befehl im Terminal (extrahiertes Verzeichnis) aus: **sh install.sh**

Nach der Installation dieser Skripte gehen Sie zu *All Apps*, wo die *Arduino-IDE* installiert ist.



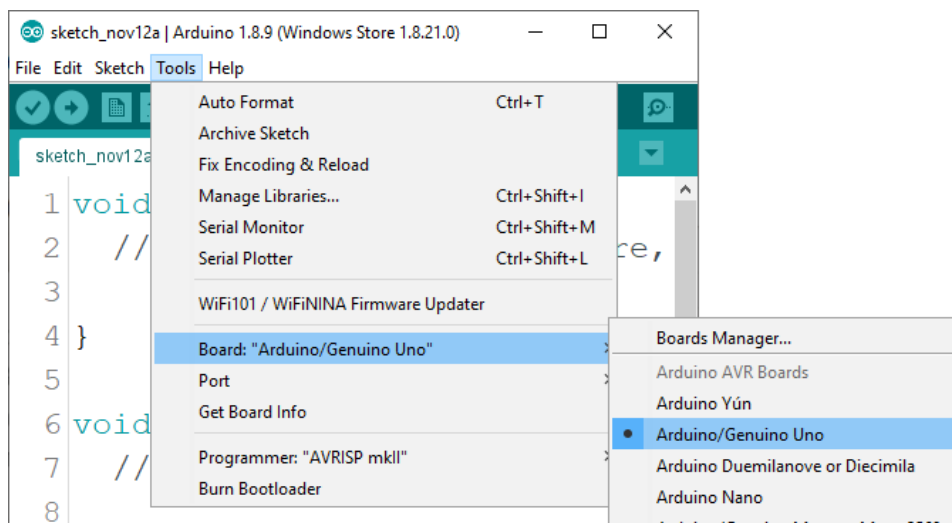
# Az-Delivery

Fast alle Betriebssysteme werden mit einem vorinstallierten Texteditor ausgeliefert (z.B. *Windows* mit *Notepad*, *Linux Ubuntu* mit *Gedit*, *Linux Raspbian* mit *Leafpad* usw.). Alle diese Texteditoren sind für den Zweck des eBooks vollkommen in Ordnung.

Zunächst ist zu prüfen, ob Ihr PC ein Arduino-Board erkennen kann. Öffnen Sie die frisch installierte Arduino-IDE, und gehen Sie zu:

*Tools > Board > {your board name here}*

*{your board name here}* sollte der *Arduino/Genuino Uno* sein, wie es auf dem folgenden Bild zu sehen kann:

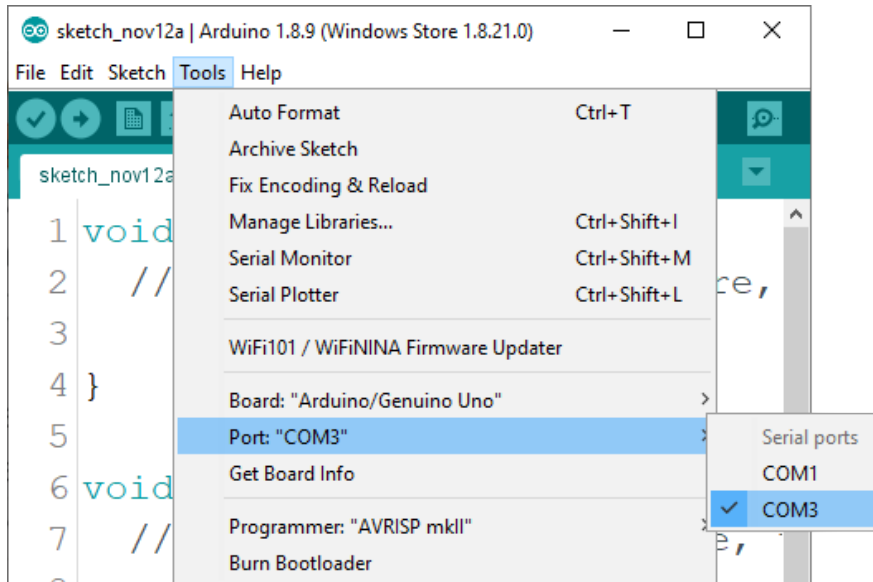


Der Port, an den das Arduino-Board angeschlossen ist, muss ausgewählt werden. Gehe zu: *Tools > Port > {port name goes here}* und wenn das Arduino-Board an den USB-Port angeschlossen ist, ist der Portname im Drop-down Menü auf dem vorherigen Bild zu sehen.



# Az-Delivery

Wenn die Arduino-IDE unter Windows verwendet wird, lauten die Portnamen wie folgt:



Für *Linux* Benutzer, ist zum Beispiel der Portname `/dev/ttyUSBx`, wobei `x` für eine ganze Zahl zwischen `0` und `9` steht.

## Wie man den Raspberry Pi und Python einrichtet

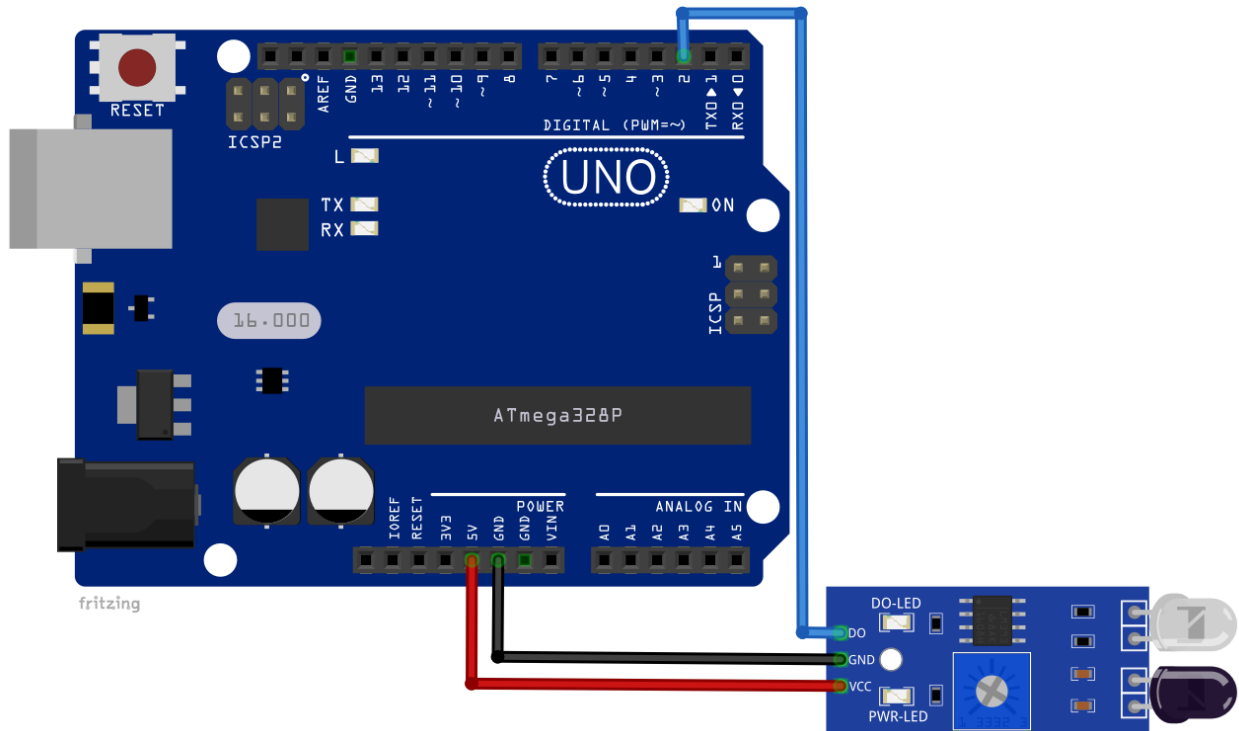
Für den Raspberry Pi muss zuerst das Betriebssystem installiert werden, dann muss alles so eingerichtet werden, dass es im *Headless*-Modus verwendet werden kann. Der *Headless*-Modus ermöglicht eine Fernverbindung zum Raspberry Pi, ohne dass ein PC-Bildschirm, eine Maus oder eine Tastatur erforderlich ist. Die einzigen Dinge, die in diesem Modus verwendet werden, sind der Raspberry Pi selbst, die Stromversorgung und die Internetverbindung. Das alles wird in dem kostenlosen eBook ausführlich erklärt:

[Raspberry Pi Quick Startup Guide](#)

Das Betriebssystem Raspbian wird mit vorinstalliertem *Python* ausgeliefert.

## Verbindung des Moduls mit dem Uno

Das Modul hat drei Pins. Die Pinbelegung ist wie folgt:



Module pin	Uno pin	Drahtfarbe
OUT	D2	Blauer Draht
GND	GND	Schwarzer Draht
VCC	5V	Roter Draht

# Az-Delivery

## Sketch-Beispiel

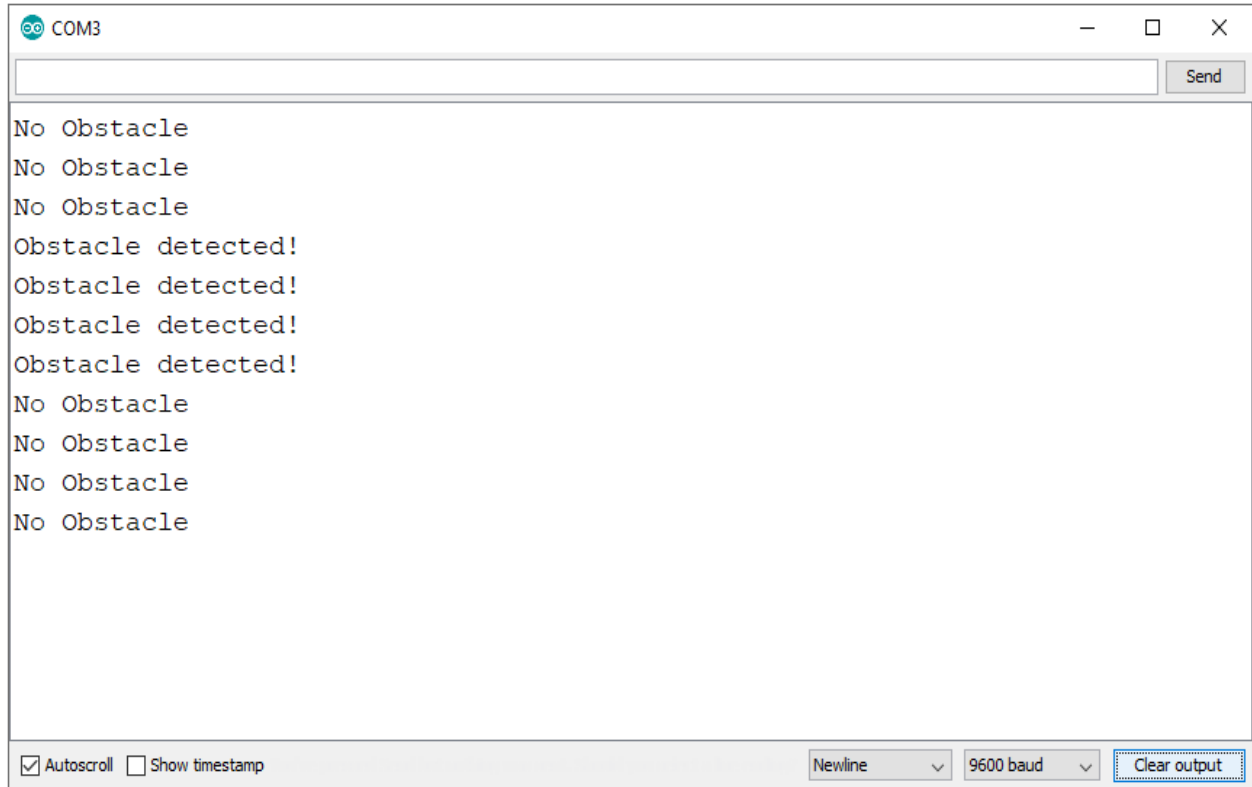
```
#define DIGITAL_PIN 2
boolean sensorOut = HIGH;

void setup() {
  Serial.begin(9600);
  pinMode(DIGITAL_PIN, INPUT);
}

void loop() {
  sensorOut = digitalRead(DIGITAL_PIN);
  if (sensorOut == LOW)
  {
    Serial.println("Obstacle detected!");
  }
  else
  {
    Serial.println("No Obstacle");
  }
  delay(1000);
}
```

# Az-Delivery

Laden Sie den Sketch in den Serial Monitor (*Tools > Serial Monitor*). Die Ausgabe sollte wie folgt aussehen:



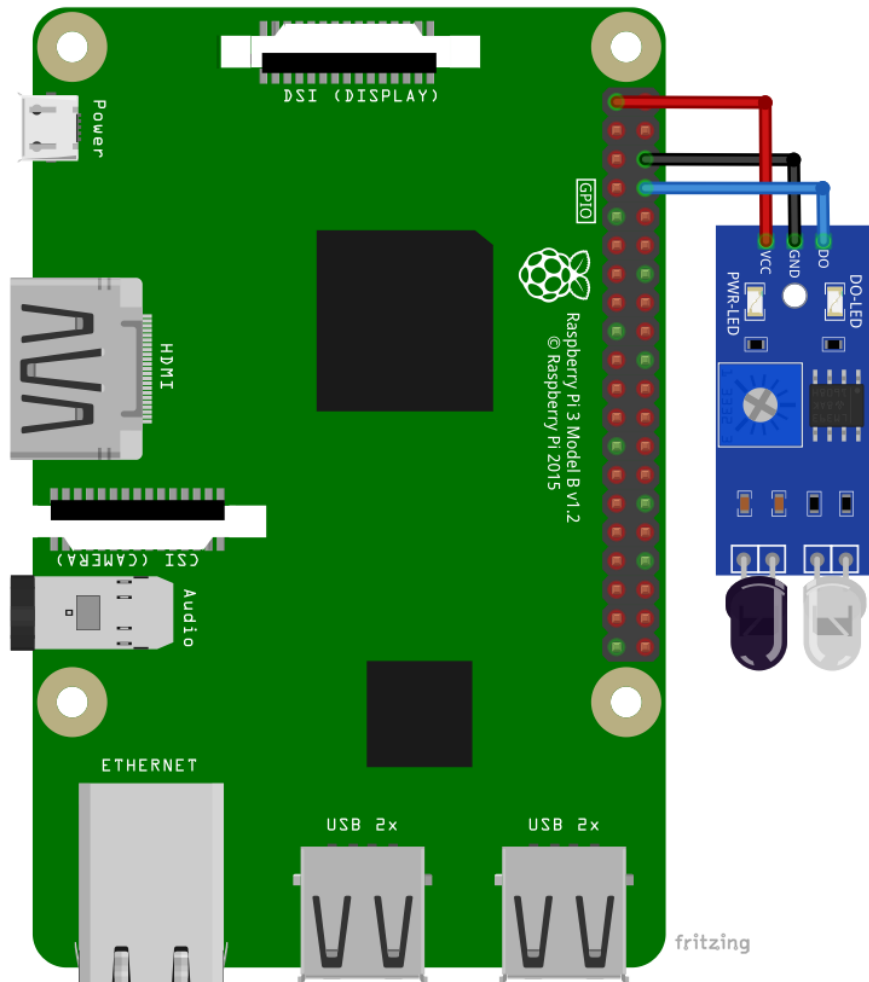
The screenshot shows the Serial Monitor window for COM3. The output text is as follows:

```
No Obstacle  
No Obstacle  
No Obstacle  
Obstacle detected!  
Obstacle detected!  
Obstacle detected!  
Obstacle detected!  
No Obstacle  
No Obstacle  
No Obstacle  
No Obstacle
```

The window also features a 'Send' button at the top right, a 'Clear output' button at the bottom right, and checkboxes for 'Autoscroll' (checked) and 'Show timestamp' (unchecked). The baud rate is set to 9600 and the line ending is set to Newline.

## Verbindung des Moduls mit dem Raspberry Pi

Verbinden Sie das Modu mit dem Raspberry Pi, wie unten abgebildet:



Module Pin	Raspberry Pi Pin	Physischer Pin	Drahtfarbe
VCC	3.3V	1	<b>Roter Draht</b>
GND	GND	6	<b>Schwarzer Draht</b>
OUT	GPIO14	8	<b>Blauer Draht</b>



## Libraries und Tools für Python

Um das Modul mit dem Raspberry Pi zu verwenden, muss die Library RPi.GPIO installiert sein. Wenn die Library bereits installiert ist, wird durch Ausführen des Installationsbefehls nur die Library auf eine neuere Version aktualisiert.

Um die Library zu installieren, öffnen Sie das Terminal und führen Sie nacheinander die folgenden Befehle aus:

```
sudo apt-get update && sudo apt-get upgrade  
sudo apt-get install python3-rpi.gpio
```

# Az-Delivery

## PythonSkript

```
import time
import RPi.GPIO as GPIO

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

DIGITAL_PIN = 14
GPIO.setup(DIGITAL_PIN, GPIO.IN)
time.sleep(2)

print('Obstacle Detection Sensor script')
print('[Press CTRL + C to end the script]!')
try: # Main program loop
    while True:
        if GPIO.input(DIGITAL_PIN)==0:
            print('Obstacle detected!')
            time.sleep(2)
        else:
            print('No obstacles!')
            time.sleep(2)
except KeyboardInterrupt:
    print('\nScript end!')
finally:
    GPIO.cleanup()
```

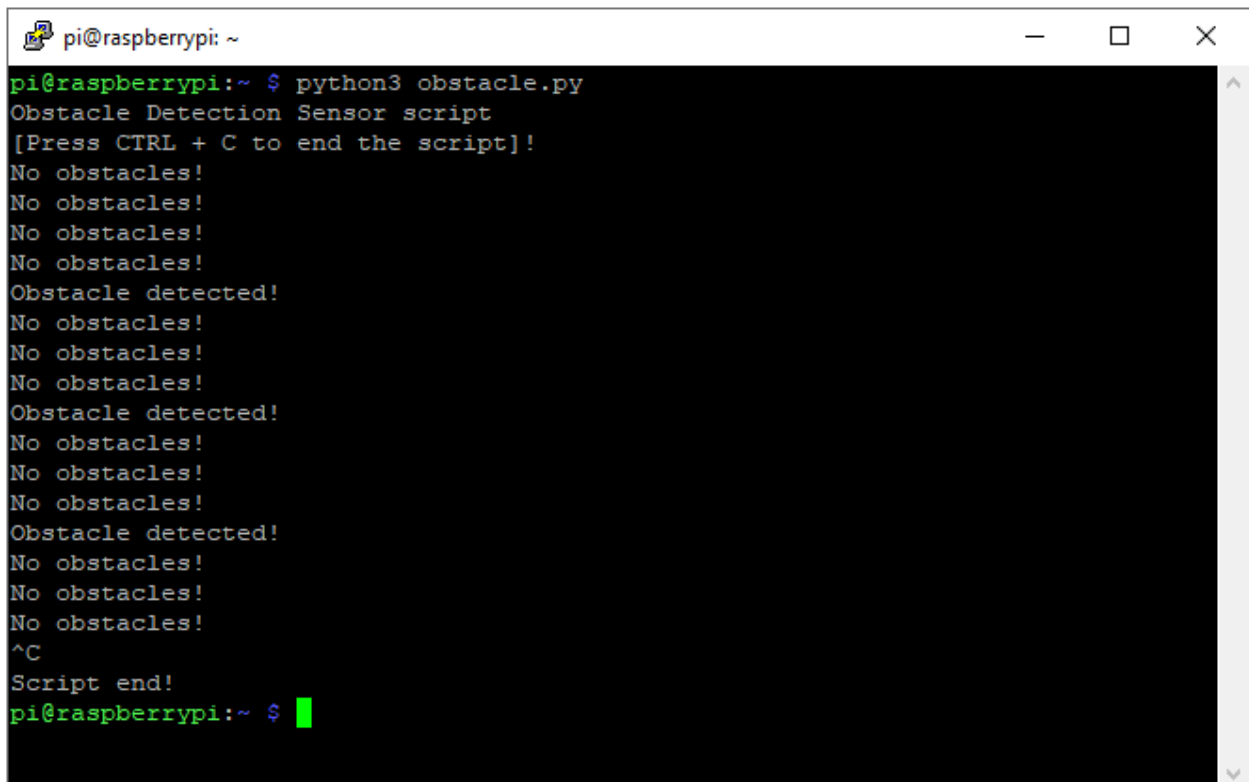


# Az-Delivery

Speichern Sie das Skript unter dem Namen *obstacle.py*. Um das Skript auszuführen, öffnen Sie das Terminal in dem Verzeichnis, in dem das Skript gespeichert ist, und führen Sie den folgenden Befehl aus:

```
python3 obstacle.py
```

Die Ausgabe sollte wie folgt aussehen:



```
pi@raspberrypi: ~  
pi@raspberrypi:~ $ python3 obstacle.py  
Obstacle Detection Sensor script  
[Press CTRL + C to end the script]!  
No obstacles!  
No obstacles!  
No obstacles!  
No obstacles!  
Obstacle detected!  
No obstacles!  
No obstacles!  
No obstacles!  
Obstacle detected!  
No obstacles!  
No obstacles!  
No obstacles!  
Obstacle detected!  
No obstacles!  
No obstacles!  
No obstacles!  
Obstacle detected!  
No obstacles!  
No obstacles!  
No obstacles!  
^C  
Script end!  
pi@raspberrypi:~ $ █
```

Um das Skript zu stoppen, drücken Sie “STRG + C” auf der Tastatur.

**Sie haben es geschafft. Sie können jetzt unser Modul für Ihre Projekte nutzen.**

# AZ-Delivery

Jetzt sind Sie dran! Entwickeln Sie Ihre eigenen Projekte und Smart-Home Installationen. Wie Sie das bewerkstelligen können, zeigen wir Ihnen unkompliziert und verständlich auf unserem Blog. Dort bieten wir Ihnen Beispielskripte und Tutorials mit interessanten kleinen Projekten an, um schnell in die Welt der Mikroelektronik einzusteigen. Zusätzlich bietet Ihnen auch das Internet unzählige Möglichkeiten, um sich in Sachen Mikroelektronik weiterzubilden.

**Falls Sie nach weiteren hochwertigen Produkten für Arduino und Raspberry Pi suchen, sind Sie bei AZ-Delivery Vertriebs GmbH goldrichtig. Wir bieten Ihnen zahlreiche Anwendungsbeispiele, ausführliche Installationsanleitungen, E-Books, Bibliotheken und natürlich die Unterstützung unserer technischen Experten.**

<https://az-delivery.de>

Viel Spaß!

Impressum

<https://az-delivery.de/pages/about-us>