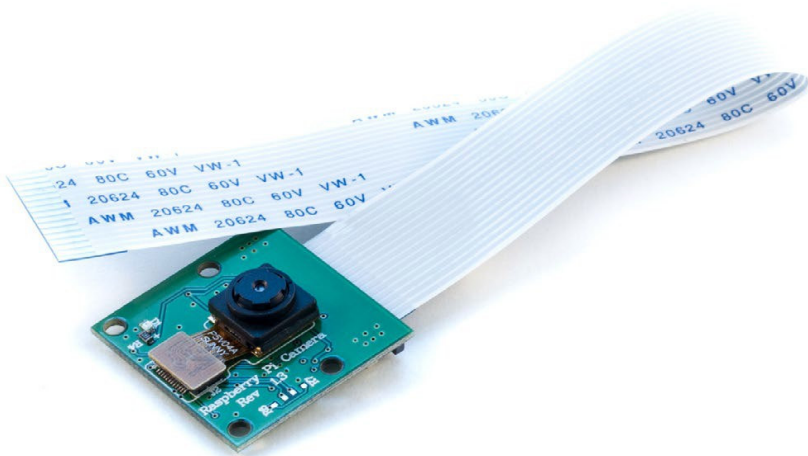


Willkommen!

Und herzlichen Dank für den Kauf unserer **AZ-Delivery Kamera v1.3 für den Raspberry Pi**! Auf den folgenden Seiten gehen wir mit dir gemeinsam die ersten Schritte von der Einrichtung bis zum Speichern deines ersten Kamerabildes. Viel Spaß!



http://flyt.it/RPI_cam_v1-3

Die Kamera besteht aus einem Controller mit austauschbarem Sensor und einem 15-poligen Flachband-Verbindungskabel, das mit dem Raspberry Pi verbunden werden muss.

Die wichtigsten Informationen in Kürze

- » Maße: 25x20x9mm
- » Gewicht: ca. 3g

- » Fixierter Fokus
- » Automatische/r Helligkeit und Weißabgleich
- » Auflösung bis 2592 x 1944px
- » Videoauflösung:
 - » 1920 x 1080 (30 Fps)
 - » 1280 x 720 (60 Fps)
 - » 640 x 480 (60 oder 90 Fps)
- » Energieversorgung und Datenübertragung über 15-poliges Flachbandkabel

Auf den nächsten Seiten findest du Informationen zur

» ***Einrichtung der Kamera***

und eine Anleitung für

» ***das Anzeigen und Speichern des Kamerabildes.***

Dieses Tutorial setzt einen Raspberry Pi mit installiertem Raspbian Betriebssystem ab Version Jessie voraus. Eine Anleitung dazu findest du auf unserer AZ-Delivery eBook-Seite:

» <http://flyt.it/eBook>

Alle Links im Überblick

Verwendete Hardware:

- » Raspberry Pi: <https://az-delivery.de/products/raspberrypi3>
- » Kamera: <https://az-delivery.de/products/raspberrycamerav1-3>
- » Flachbandkabel: <https://az-delivery.de/.../15cm-ersatz-flexkabel-fur-raspberry-pi-kamera-und-das-raspberry-pi-display>

Verwendete Software:

- » Raspbian: <https://www.raspberrypi.org/downloads/raspbian/>
- » Thonny (Python IDE): <http://thonny.org/>
- » PiCamera-Bibliothek: <http://picamera.readthedocs.org/>

Raspberry Pi Tutorials, Beispiele, Referenz, Community:

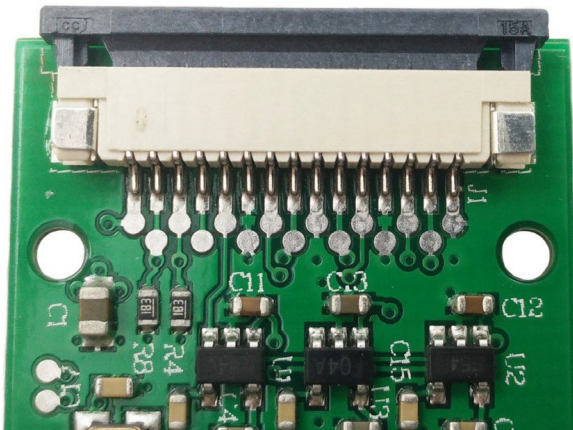
- » <https://www.raspberrypi.org/documentation/>
- » <https://www.raspberrypi.org/education/>

Interessantes von AZ-Delivery

- » Alle Tutorial eBooks:
<https://az-delivery.de/pages/ebooks-und-dokumentationen-zum-kostenlosen-download>
- » Raspberry Pi Zubehör:
<https://az-delivery.de/collections/raspberry-pi-zubehor>
- » AZ-Delivery G+Community:
<https://plus.google.com/communities/115110265322509467732>
- » AZ-Delivery auf Facebook:
<https://www.facebook.com/AZDeliveryShop/>

Einrichtung der Kamera

Bevor wir mit der Programmierung beginnen, muss zunächst die Hardwarezusammenstellung stimmen. Wir benötigen dazu den Raspberry Pi, das Kamera-Modul und ein Verbindungskabel. Deine **AZ-Delivery Kamera v1.3** wird standardmäßig mit bereits verbundenem Kabel geliefert. Falls du z.B. mit einem längeren Kabel arbeiten möchtest, lege das Modul mit der Linse nach unten auf eine saubere Fläche und ziehe den schwarzen Balken am Anschluss an den Seiten vorsichtig nach außen, bis er locker ist.

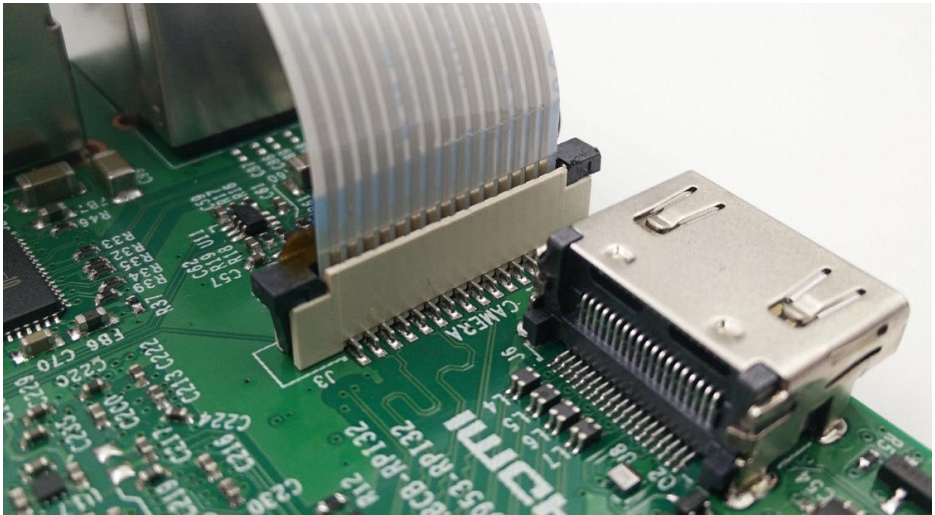


Führe anschließend ein Kabelende mit der blauen Seite nach oben gerade in den Schlitz zwischen dem Balken und dem hellen Gehäuseteil, bis du einen Widerstand spürst. Drücke dann den Balken vorsichtig wieder fest, bis er einrastet:



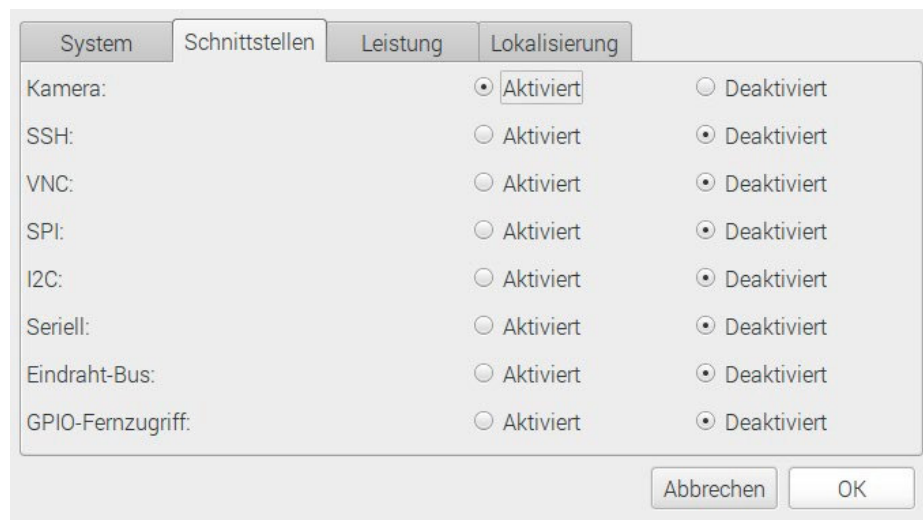
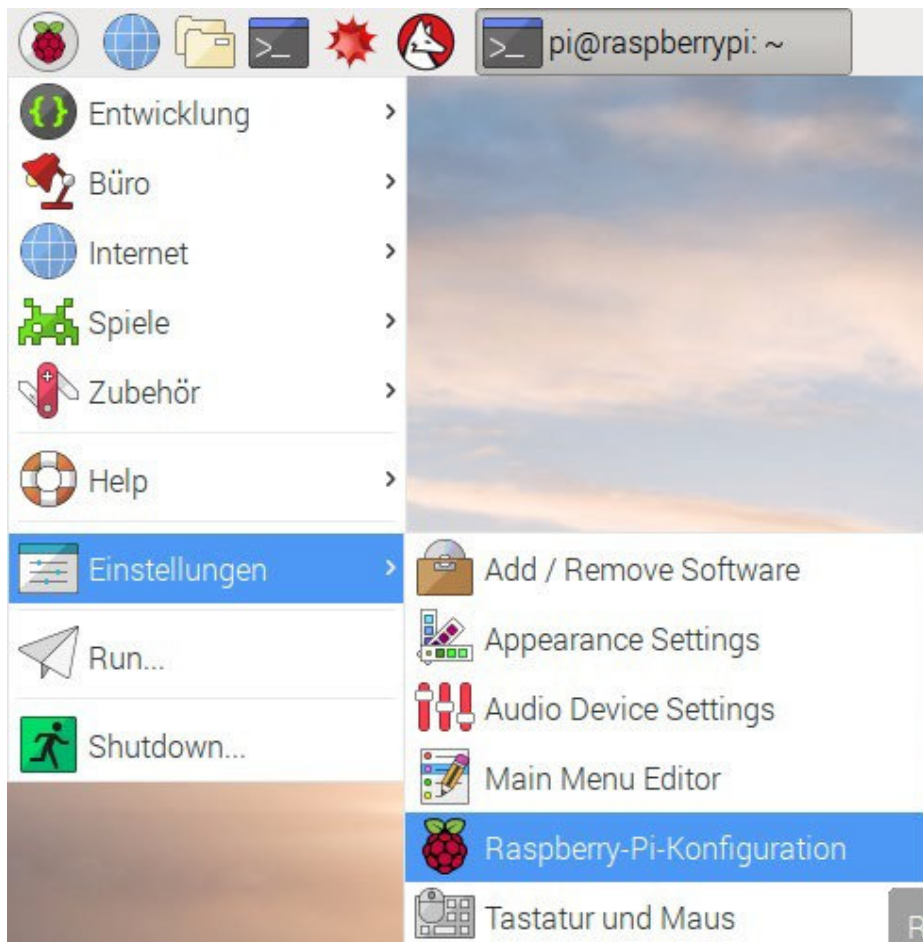
Das gleiche machst du mit der anderen Seite an dem mit CAMERA beschrifteten Anschluss deines Raspberry Pi. Dabei muss die blaue Seite Richtung Netzwerk- und die Kontaktseite in Richtung Micro-USB-Anschluss zeigen.

Stelle zuvor sicher, dass dein Raspberry Pi von seiner Stromquelle getrennt wurde! Eine unbeabsichtigte Berührung mehrerer Kontakte kann ansonsten zu Schäden am Board führen.



Wenn die Kamera sicher und fest mit dem Raspberry Pi verbunden ist, kannst du deinen Controller wieder anschließen und starten.

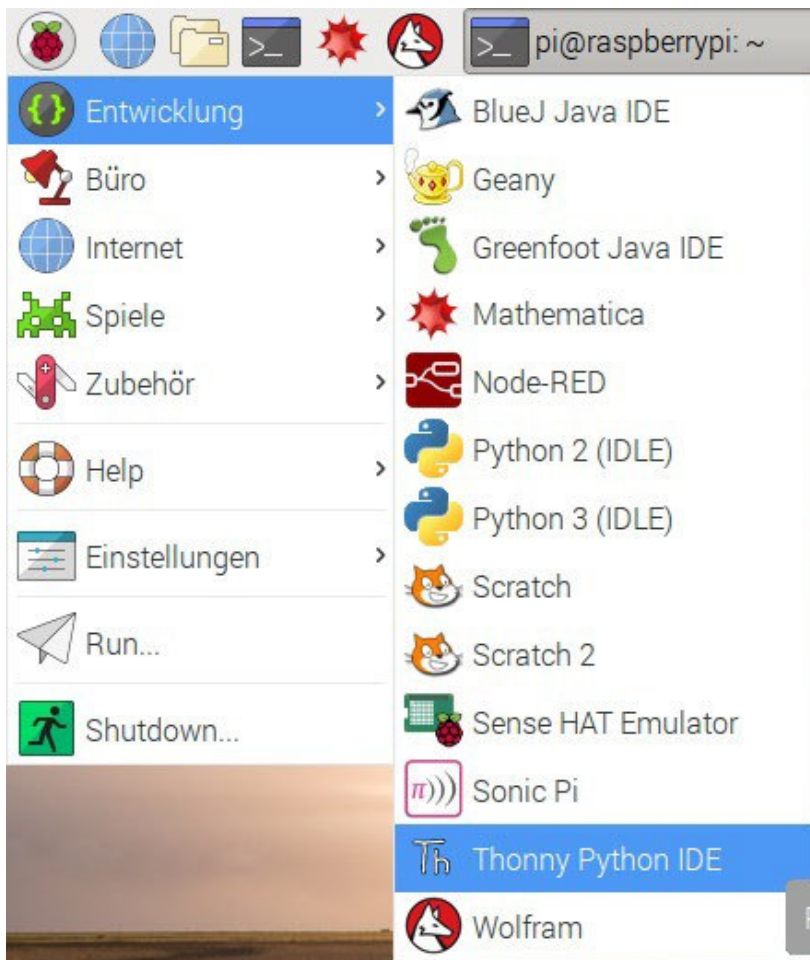
Bei einem frisch installierten Raspbian-System sind in der Regel sämtliche Schnittstellen deaktiviert, auch die für die Kamera. Um das zu ändern, öffne im Bereich **"Einstellungen"** die Anwendung **"Raspberry-Pi-Konfiguration"**. Unter dem Reiter **"Schnittstellen"** stellst du nun die Option **"Kamera:"** auf **"Aktiviert"**.



Anzeigen und Speichern des Kamerabildes

An dieser Stelle lernst du, wie du mithilfe der beliebten Scriptsprache **Python** Zugriff zur Kamera erlangst und nach einer definierten Zeitspanne einen Frame als Bild abspeicherst.

Bei Raspbian ist Python bereits vorinstalliert, ebenso (seit Juni 2017) die **Thonny Python IDE**, die wir in diesem Tutorial auch verwenden.



Starte die IDE und schreibe folgenden Code in den Editor-Bereich. Speichere ihn als *.py-Datei in ein Verzeichnis deiner Wahl.

```
1  import time
2  from picamera import PiCamera
3
4  def main():
5      camera = PiCamera()
6      camera.start_preview()
7      time.sleep(10)
8      camera.capture("/home/pi/Pictures/camera.jpg")
9      camera.stop_preview()
10
11  if __name__ == "__main__":
12      main()
```

Erklärungen:

`import time` lädt die Bibliothek, mit der wir über den Befehl `time.sleep(10)` das Abarbeiten des Codes an einer bestimmten Stelle für zehn Sekunden pausieren.

Von `picamera` beziehen wir die Methoden zum Vollbild-Streamen `camera.start_preview()` und Speichern `camera.capture("/.../camera.jpg")` des Kamerabildes.

Der Befehl `camera.stop_preview()` am Ende der `main()`-Methode ist ebenfalls wichtig und sollte immer nach einem **`start_preview()`**-Befehl dazugeschrieben werden, da die Vollbildansicht der Kamera ansonsten nicht mehr geschlossen wird und du deinen Raspberry Pi neu starten musst.



Hast du den Code richtig eingegeben und gespeichert, führe ihn über das dreieckige **Play**-Symbol aus und erfreue dich an deinem Kamerabild. Du hast nun ca. zehn Sekunden Zeit, dir ein Motiv zu suchen, welches anschließend als JPG-Datei auf deinem Raspberry Pi landet.



(Unsere Kamera hat gerade ihre Basis betrachtet.)

Herzlichen Glückwunsch!

Du hast das Tutorial erfolgreich durchlaufen, deine Kamera angeschlossen und mit ihr dein erstes Foto geschossen! Ab jetzt heißt es lernen und ausprobieren. Schau dir dazu am besten die Dokumentation der **PiCamera**-Bibliothek an.

Wenn du nun auch in Infrarot sehen möchtest oder einfach andere tolle Hardware suchst, wirst du wie immer fündig bei deinem Online-Shop auf:

<https://az-delivery.de>

Viel Spaß!

Impressum

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