

Project PI



Project PI Protokolle

Autor:
Johannes Ucel

1 Sensoren – Tests

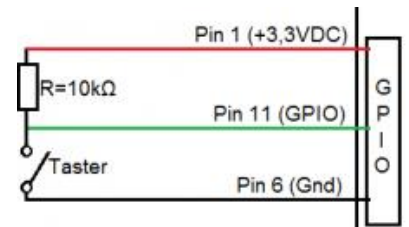
Die Ansteuerung der Ports erfolgte mit dem Command-Line-Tool „pigpio“, der Python Konsole bzw. Python Code oder der Shell.

Folgende Sensoren wurden getestet:

1.1.1 Taster

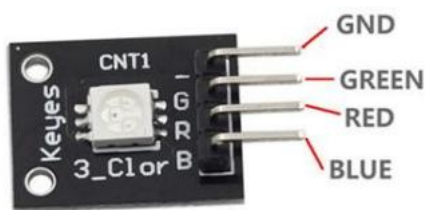
Beim Testen wurde ein Pullup – Widerstand verwendet, wodurch am GPIO- Port immer „high“-Signal (entspricht 1) anliegt. Wird er nun mit dem Ground – Port verbunden, ändert sich der Zustand auf „low“ (entspricht 0)

Der Taster wurde erfolgreich getestet.



1.1.2 Kombi LED

Es wurde sowohl eine SMD (Surface Mount Device) – Kombi LED, als auch eine DIP (Dual In-line Package) – Kombi LED getestet.

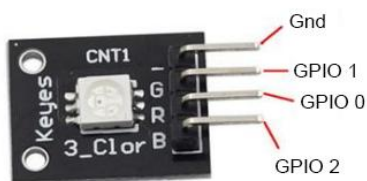


SMD Package



DIP Package

Beide LEDs wurden mit den gleichen Ports verbunden:



SMD Package



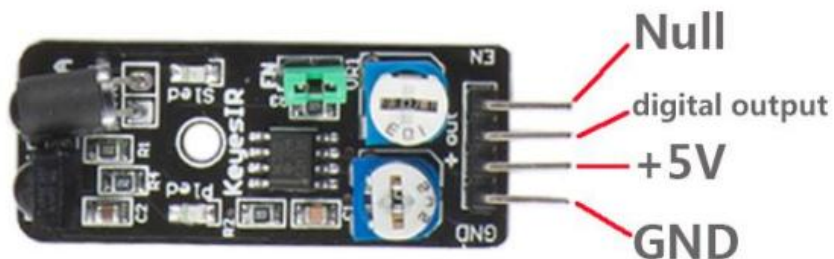
DIP Package

Über die Konsolen und dem „pigpio“ – Tool, wurden verschiedene RGB – Farben erfolgreich getestet.



1.1.3 Annäherungssensor

Der Annäherungssensor verwendet die Infrarot – Reflektion, um Hindernisse zu erkennen. Sollte kein Gegenstand davor sein, kann er kein Signal empfangen. Falls ein Gegenstand davor ist, wird das Infrarotlicht reflektiert und der Receiver kann ein Signal empfangen.

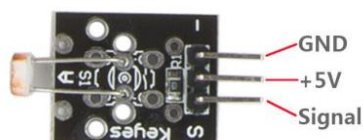


Der digitale Output wurde mit dem GPIO 0 – Port verbunden.

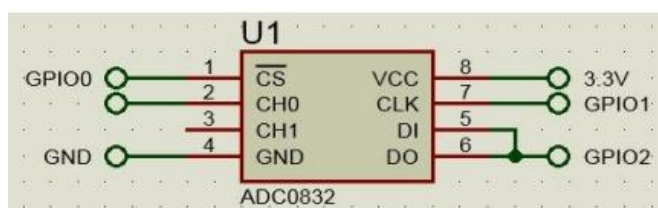
Die restlichen Anschlüsse wurden wie oben angeführt mit den Ports verbunden.

1.1.4 Helligkeitssensor

Der Helligkeitssensor misst die Helligkeit und gibt einen digitalen Wert aus.

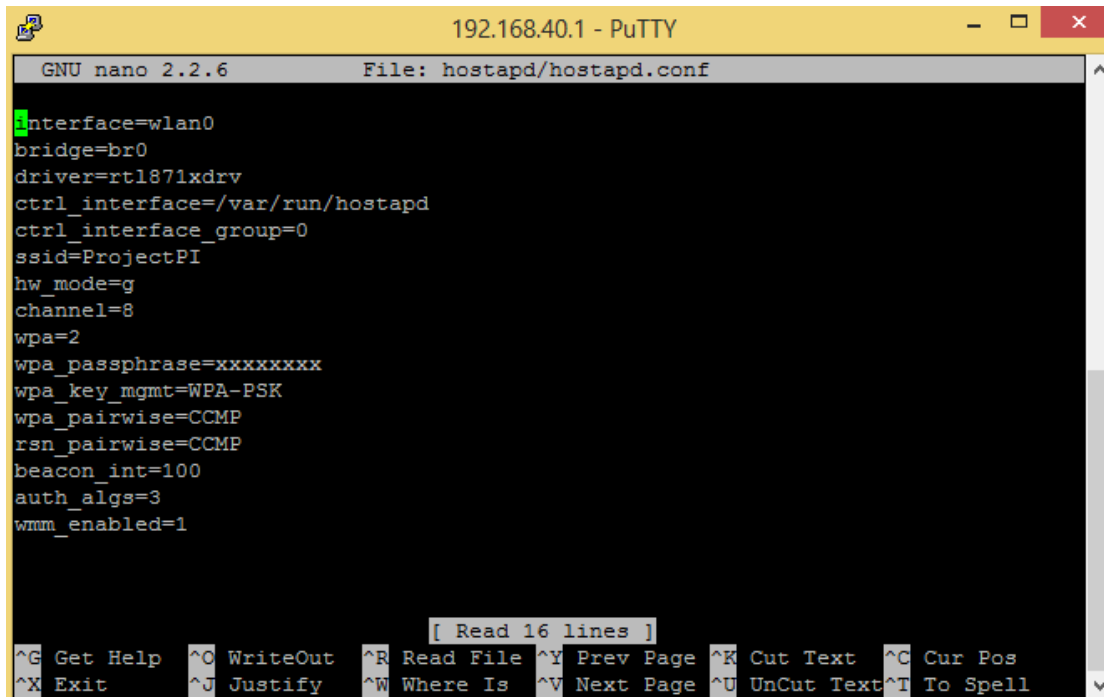


Die Ports wurden folgendermaßen verbunden:



1.1.5 WLAN – Kommunikation

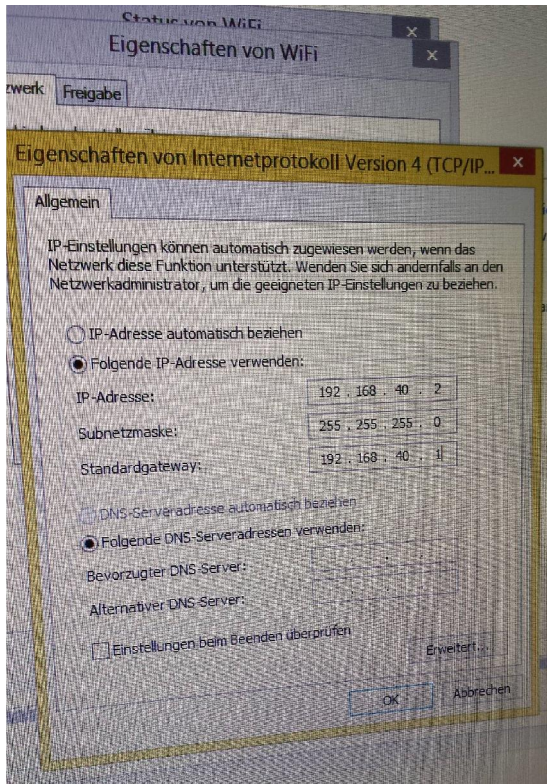
Die WLAN – Kommunikation wurde mit einem Realtek WiFi Dongle realisiert. So kann man sich mit dem Smartphone oder anderen Geräten (Notebook) mit dem Raspberry verbinden.



```
192.168.40.1 - PuTTY
GNU nano 2.2.6 File: hostapd/hostapd.conf
interface=wlan0
bridge=br0
driver=rtl871xdrv
ctrl_interface=/var/run/hostapd
ctrl_interface_group=0
ssid=ProjectPI
hw_mode=g
channel=8
wpa=2
wpa_passphrase=xxxxxxxx
wpa_key_mgmt=WPA-PSK
wpa_pairwise=CCMP
rsn_pairwise=CCMP
beacon_int=100
auth_algs=3
wmm_enabled=1

[ Read 16 lines ]
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

Über die SSH – Verbindungen können Konfigurationen durchgeführt werden, auf Daten (SFTP) zugegriffen werden etc.



1.2 Kamera

Mit dem folgenden Code kann sowohl ein Foto als auch ein Video mit der „Pi-Cam“ aufgenommen werden.

```
__author__ = 'Johannes Ucel'  
import picamera  
from time import sleep  
camera = picamera.PiCamera()  
camera.capture('image.jpg')  
camera.start_recording('video.h264')  
sleep(5)  
camera.stop_recording()
```