

Raspberry Pi Grundlagen

I. Raspbian installieren / SD Karte vorbereiten

1. Raspbian Image (www.raspberrypi.org/downloads) herunterladen, entpacken
2. SD Formatter (www.sdcard.org/downloads/formatter_4) herunterladen, installieren, starten
Richtiges Wechselmedium / SD Karte auswählen
Optionen „Format Type“ auf „Quick“ und „Format Size Adjustment“ auf „On“
„Format“ klicken
3. Win32 Disk Imager (<http://sourceforge.net/projects/win32diskimager/>) herunterladen, entpacken, starten
Richtiges Wechselmedium / SD Karte auswählen
Unter „Image File“ das entpackte Image auswählen
„Write“ klicken
4. SD Karte nach erfolgreichem Schreibvorgang auswerfen und in den RPi stecken, booten

II. Raspbian konfigurieren nach Boot

- | | | |
|--|--------------------------------|------------------------------------|
| a) <u>raspi-config</u> | b) <u>Update Software</u> | c) <u>Update Firmware / Kernel</u> |
| 1. <i>Expand Filesystem</i> | 1. <i>sudo apt-get update</i> | 1. <i>sudo rpi-update</i> |
| 2. <i>Change User Password</i> | 2. <i>sudo apt-get upgrade</i> | 2. <i>sudo shutdown -r now</i> |
| 3. <i>Enable Boot to Desktop</i> | 3. <i>sudo shutdown -r now</i> | |
| 4. <i>Internationalisation Options</i> | | |
| 5. <i>Neustarten</i> | | |

III. Vorbereitungen

- a) I²C / SPI Bus
- Software installieren
sudo apt-get update
sudo apt-get install i2c-tools python-smbus git-core python-serial python-dev python-setuptools
 - Umgebung vorbereiten
 1. */etc/modprobe.d/raspi-blacklist.conf* editieren, z.B. *sudo vi /etc/modprobe.d/raspi-blacklist.conf*
Beide „blacklist“ auskommentieren und speichern:
#blacklist spi-bcm2708
#blacklist i2c-bcm2708
 2. */etc/modules* editieren, z.B. *sudo vi /etc/modules*
Am Ende um folgende Einträge erweitern und speichern:
i2c-bcm2708
spi-bcm2708
i2c-dev
 3. User pi zur Gruppe i2c, spi und gpio hinzufügen: *sudo usermod -a -G i2c,spi,gpio pi*
 4. Reboot: *sudo shutdown -r now*

b) Serielle Schnittstelle vorbereiten

- rpi-serial-console installieren
sudo wget https://raw.githubusercontent.com/lurch/rpi-serial-console/master/rpi-serial-console -O /usr/bin/rpi-serial-console
sudo chmod +x /usr/bin/rpi-serial-console
- Serielle Konsole abschalten
sudo rpi-serial-console disable
sudo shutdown -r now

c) wiringpi2 installieren

- Library installieren
cd ~
git clone git://git.drogon.net/wiringPi
cd wiringPi
./build
- Library testen:
gpio -v
gpio readall

d) wiringpi2-python installieren

- Library installieren

cd ~

```
git clone https://github.com/Gadgetoid/WiringPi2-Python.git
```

```
cd WiringPi2-Python
```

```
sudo python setup.py install
```

- Library testen:

```
sudo python
```

```
import wiringpi2
```

```
wiringpi2.piBoardRev()
```

```
exit()
```

e) spidev installieren

- Library installieren

cd ~

```
git clone git://github.com/doceme/py-spidev
```

```
cd py-spidev/
```

```
sudo python setup.py install
```

f) Wichtig

- Maximale Belastbarkeit GPIO beachten
- Logik arbeitet mit 3,3V, keine 5V Toleranz
- Änderung Pin Belegung I²C bei RPi Rev 2
- RPi herunterfahren: **sudo shutdown -h now**
- RPi neustarten: **sudo shutdown -r now**

IV . Erstes Experiment: Digitaler Ein und Ausgang

a) BOM

D0: LED

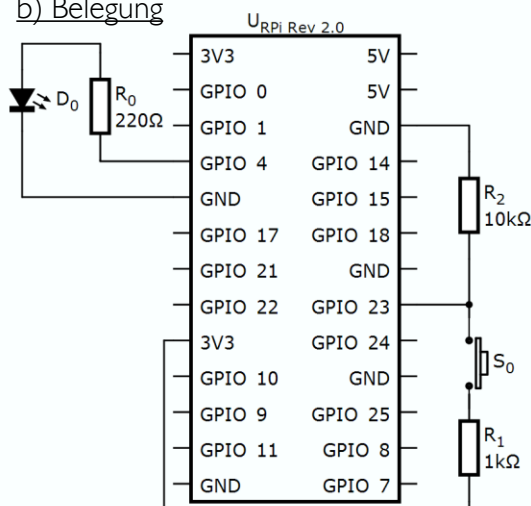
R0: 220 Ω

S0: Schalter

R1: 1k Ω

R2: 10k Ω

b) Belegung



c) Quellcode

```
# Datei bitte 3_DI_DO.py benennen
```

```
import wiringpi2
```

```
# wiringpi2 Library laden
```

```
from time import sleep
```

```
# sleep aus der time Library laden
```

```
wiringpi2.wiringPiSetupGpio()
```

```
# wiringpi2 initialisieren mit GPIO Nummerierung
```

```
wiringpi2.pinMode(4,I)
```

```
# GPIO4 in Ausgangsmodus schalten
```

```
wiringpi2.pinMode(23,O)
```

```
# GPIO23 in Eingangsmodus schalten
```

```
while True:
```

```
# Endlosschleife
```

```
    schalter = wiringpi2.digitalRead(23)
```

```
# GPIO23 lesen und in Variable schalter schreiben
```

```
    if schalter:
```

```
# Falls schalter wahr ist, schalte die LED an
```

```
# und gebe „Schalter gedruickt“ aus
```

```
        wiringpi2.digitalWrite(4,I)
```

```
        print "Schalter gedruickt"
```

```
    else:
```

```
# sonst schalte die LED aus und gebe
```

```
# „Schalter nicht gedruickt“ aus
```

```
        wiringpi2.digitalWrite(4,O)
```

```
        print "Schalter nicht gedruickt,,"
```

```
        sleep(0.25)
```

```
# pausiere ¼ Sekunde
```

```
d) Programm starten
```

```
sudo python 3_DI_DO.py
```

GPIO Pinbelegung RPi Rev. 2

Pin	Label	Color
1	+3.3V	Orange
2	GPI00(SDA1)	Blue
3	GPI01(SCL1)	Blue
4	GPI04(GPCLK0)	Green
5	GND	Black
6	GPI017	Green
7	GPI027	Blue
8	GPI022	Green
9	+3.3V	Orange
10	GPI010(MOSI)	Pink
11	GPI09(MISO)	Pink
12	GPI011(SCLK)	Pink
13	GND	Black
14	+5V	Red
15	+5V	Red
16	GND	Black
17	GPI014(TXD)	Yellow
18	GPI015(RXD)	Yellow
19	GPI018(PCM_CLK)	Green
20	GND	Black
21	GPI023	Green
22	GPI024	Green
23	GND	Black
24	GPI025	Green
25	GPI08(CE0)	Pink
26	GPI07(CE1)	Pink