

RaspberryPI Fernsteuerung Dokumentation

Alexander von Leliwa
Bernhard Schmitt

8. Januar 2014

Inhaltsverzeichnis

1	Beispiele	3
1.1	quellcode	3
1.2	links	4
1.3	fabige frames	5
2	Einrichten des Raspberry PI	5
2.1	Betriebssystem	5
2.2	WLAN-Einrichten	5
2.2.1	Stick erkennen	5
2.2.2	Schlafmodus deaktivieren	5
2.2.3	Verbindung herstellen	5
3	Hardware anschluss	6
4	xml Schnittstellendatei	6
5	PHP Benutzerplattform	6
6	Backend (Python Steuerung)	6
6.1	Chrome-Job	6
6.2	Temperaturfühler initialisieren	6
6.3	Hauptprogramm	7

1 Beispiele

1.1 quellcode

```
#!/usr/bin/env python3
#Eine Wirkung, z.B. Pin schalten, wird ausgefuehrt

#Initialisierung

def wk_exe(art, pinnr, typ, wert):
    # art 1 --> GPIO
    if art == 1:
        import RPi.GPIO as GPIO
        GPIO.setmode(GPIO.BOARD)
```

```
$ ifconfig
```

Listing 1: Wlan einbinden

```
# erstellt die interfaces Datei
def Datei(name, pw):
    datei = open("/home/bernhard/Dokumente/Phyton/interfaces", "w")
    datei.write("auto lo \n")
    datei.write("iface lo inet loopback\n\n")
    datei.write("iface eth0 inet dhcp\n")
    datei.write("iface default inet dhcp\n\n")
    datei.write("auto wlan0\n")
    datei.write("iface wlan0 inet dhcp\n")
    datei.write("wpa-ssid ")
    datei.write(name)
    datei.write("\n")
    datei.write("wpa-psk ")
    datei.write(pw)
    datei.close()

#Hauptmenue
name = input("Geben Sie ihren Wlan Namen ein: ")
pw = input("Geben Sie ihr Wlan Passwortein: ")
print(name, pw)
Datei(name, pw)
```

oder mit farbe

Listing 2: Wlan einbinden mit farbe

```
# erstellt die interfaces Datei
def Datei(name, pw):
    datei = open("/home/bernhard/Dokumente/Phyton/interfaces", "w")
    datei.write("auto lo \n")
    datei.write("iface lo inet loopback\n\n")
    datei.write("iface eth0 inet dhcp\n")
    datei.write("iface default inet dhcp\n\n")
    datei.write("auto wlan0\n")
    datei.write("iface wlan0 inet dhcp\n")
    datei.write("wpa-ssid ")
    datei.write(name)
    datei.write("\n")
    datei.write("wpa-psk ")
    datei.write(pw)
    datei.close()

#Hauptmenue
name = input("Geben Sie ihren Wlan Namen ein: ")
pw = input("Geben Sie ihr Wlan Passwortein: ")
print(name, pw)
Datei(name, pw)
```

1.2 links

Auf dieser Seite befindet sich ein Hyperlink auf die folgende Seite.

Hier befindet sich das Ziel. In die große, weite Welt geht es hier.

1.3 fabige frames

```
bla test bla
```

2 Einrichten des Raspberry PI

2.1 Betriebssystem

Wir verwenden das Betriebssystem Raspbian und installieren es über das Programm NOOBS da dieses eine einfach Erstinstallation ermöglicht. Weitere Informationen und Versionen gibt es auf dieser Seite

2.2 WLAN-Einrichten

Hilfreiche Links: Datenreise.

2.2.1 Stick erkennen

Verwendet wird der EDIMAX Wireless Adapter. Dieser wird nach dem einstecken vom Raspi automatisch erkannt. Testen kann man dies mit dem Befehl:

```
$ ifconfig
```

nun sollte auf dem Bildschirm eine Aufzählung der verschiedenen Netzwerkschnittstellen kommen. Darunter sollte auch das WLAN auftauchen.

2.2.2 Schlafmodus deaktivieren

Nun sollten wir den Stromsparmodus des Sticks deaktivieren. Um dies zu tun, muss eine Konfigurationsdatei für den Treiber angelegt werden:

```
$ sudo nano /etc/modprobe.d/8192cu.conf
```

Diese Datei bekommt folgenden Inhalt:

```
options 8192cu rtw_power_mgnt=0 rtw_enusbss=0
```

2.2.3 Verbindung herstellen

Um eine Verbindung herzustellen müssen wir eine Datei editieren:

```
$ sudo nano /etc/network/interfaces
```

den Inhalt wie folgt anpassen:

```
aus unserem Raspi kopieren
```

Abschließend muss noch die Änderung geändert werden:

```
$ sudo service networking restart
```

Das Programm das dieses Automatisch erledigt ist hier (muss noch verlinkt werden) erklärt.

3 Hardware anschluss

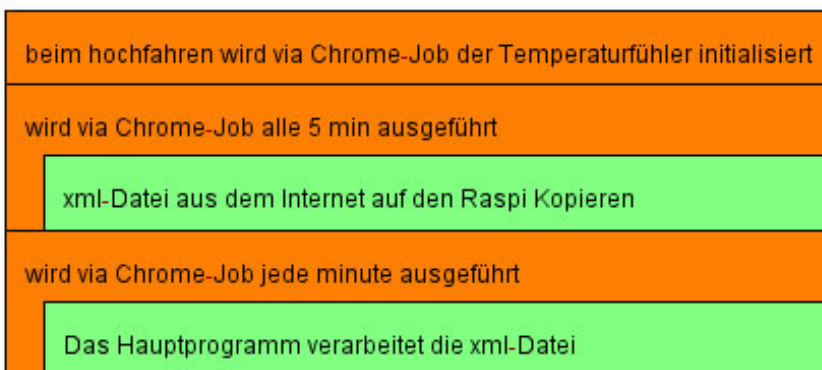
4 xml Schnittstellendatei

5 PHP Benutzerplattform

6 Backend (Python Steuerung)

Die grobe Struktur

Raspberry PI



6.1 Chrome-Job

6.2 Temperaturfühler initialisieren

Um den Temperaturfühler zu initialisieren müssen wir in der Konsole folgende Befehle eingeben:

```
sudo modprobe w1-gpio  
sudo modprobe w1-therm
```

nun ist wurde automatisch ein Ordner angelegt welcher die Temperatur beinhaltet. er ist wie folgt zu finden:

```
cd /sys/bus/w1/devices/serial_number (bsp: 10-000802824e58)
```

der letzte Ordner entspricht der Serien Nummer des verwendeten Temperaturfühlers. Der Inhalt der Datei sieht in etwa so aus:

```
2d 00 4b ff ff 08 10 fe : crc=fe YES
2d 00 4b ff ff 08 10 fe t=22250
```

t=xxxxx gibt und hierbei die Temperatur an nur das der Wert noch durch 1000 geteilt werden muss. In diesem Beispiel haben wir also 22,250 C

hilfreiche links:

Sensor einlesen mit Datei ausgabe: hier

Sensor einfaches Einlesen mit manueller Seriennummer eingabe: hier

6.3 Hauptprogramm

Listing 3: Hauptprogramm

```
#!/usr/bin/env python3
import xmleinlesen
import bedingung-pruefen
import wirkung-ausfuehren

#xml einlesen
root=xmleinlesen.einlesen()

#Zaehlen Wieviele Programme sind Aktiv?
AnzProgramme = 0
for child in root[0]:
    AnzProgramme=AnzProgramme+1
AnzProgramme=AnzProgramme-1 #-1 da Benutzer ID auch child von root[0] ist
#print('Anzahl Programme: ',AnzProgramme)

#Fuer alle Programme wird nun gleich weiterverfahren
i = 0
while (i < AnzProgramme):
    i=i+1
    verglicher=root[0][i][1].text
    vergleich = int(verglicher) #die string wird in int umgewandelt fuer den Vergleich

    #Ueberpruefen ob das Programm i aktiv ist
    if vergleich == 1:
        negativwert = 0 #Prueft, ob mindestens eine Anweisung erfuehlt ist
        #Wieviele Anweisungen gibt es fuer dieses Programm
        AnzAnweisung = 0
        for child in root[0][i]:
            AnzAnweisung=AnzAnweisung+1
        AnzAnweisung=AnzAnweisung-2 #-2 da Programmname und aktivierung auch child von programm[0] ist
        #print('Anzahl Anweisungen fuer Programm ',i,' : ',AnzAnweisung)
        #print('Programm ',i,' ist aktiv')

    #Fuer alle Anweisungen wird nun gleich weiterverfahren
    j = 1
    while (j < AnzAnweisung+1): #da wir die erste anweisungen erst bei [2] haben
        j=j+1
```

```

#Wieviele Bedingungen haben wir in dieser Anweisung
AnzBedingung = 0
for child in root[0][i][j][0]:
    AnzBedingung=AnzBedingung+1
#print('Anzahl Bedingung fuer Programm ',i,' und Anweisung ',j-1,' : ',AnzBedingung)

#Fuer jede Bedingung wird nun wie gleich weiterverfahren
n=0
aktiv=1
while (n<AnzBedingung):
    objekt = root[0][i][j][0][n][0].text #der Wert wird objekt uebergeben
    objekt = int(objekt)
    typ = root[0][i][j][0][n][1].text #Wert wird typ uebergeben
    typ = int(typ)
    wert = root[0][i][j][0][n][2].text #Wert wird wert uebergeben
    wert = int(wert)
    print(objekt, typ, wert)
    aktivieren=bedingung_pruefen.bd_check(objekt,typ,wert)
    aktiv=aktivieren*aktiv
    n=n+1
    print('fuer Programm ',i,' und Anweisung ',j-1,' und Bedingung ',n,' :')
    print('aktiv:', aktivieren)

#Wirkung fuer jede Anweisung wenn Anweisung aktiv
print('fuer Programm ',i,' und Anweisung ',j-1,' Aktiv? :',aktiv)

#Anzahl Wirkungen
AnzWirkungen = 0
for child in root[0][i][j][1]:
    AnzWirkungen=AnzWirkungen+1
#print('Anzahl Wirkung fuer Programm ',i,' und Anweisung ',j-1,' : ',AnzWirkungen)

#fuer jede Wirkung wird der GPIO gesteuert
m=0
if aktiv == 1:
    while (m < AnzWirkungen):
        art = root[0][i][j][1][m][0].text #Wert wird art uebergeben
        art = int(art)
        pinnr = root[0][i][j][1][m][1].text #Wert wird pinnr uebergeben
        pinnr = int(pinnr)
        typ = root[0][i][j][1][m][2].text #Wert wird typ uebergeben
        typ = int(typ)
        wert = root[0][i][j][1][m][3].text #Wert wird wert uebergeben
        wert = int(wert)
        wirkung_ausfuehren.wk_exe(art, pinnr, typ, wert)
        m=m+1
        negativwert=negativwert+1

if negativwert < 1:
    art = int(root[0][i][2][1][0][0].text) #Wert wird art uebergeben
    pinnr = int(root[0][i][2][1][0][1].text) #Wert wird pinnr uebergeben
    typ = int(root[0][i][2][1][0][2].text) #Wert wird typ uebergeben
    wert = 1
    wirkung_ausfuehren.wk_exe(art, pinnr, typ, wert)

#Programm i ist inaktiv
else:
    print('Programm ',i,' ist inaktiv')
#print ('Anzahl Programme: ',i)

```