

Anleitung Jukebox Kinder (<https://github.com/MiczFlor/RPi-Jukebox-RFID>) mit Buttons und HifiBerry miniAmp

Version 1.0 12.12.2017

Was wird benötigt:

[Pi 3](#) 35€

[HifiBerry miniAMP](#) 27€

[Netzteil mit genug Leistung \(3A\)](#) 11€

[Speicherkarte nach Bedarf Class 10](#) 11€

[Bastelplatinen](#) 16€

[Steckbrückenkabel](#) 7€

[MicroUSB Buchse](#) 3,50€

[PowerBank als USV und Batterie](#) 23€

[Pins zum löten](#) 7€

[RFID Lesegerät USB](#) 14€

[RFID Karten](#) 13€

[Taster](#) 7,30€

[Visaton 3W Lautsprecher](#) 2 Stück zu je 10€

Es werden zusätzlich noch Widerstände 10kOhm und 1kOhm benötigt.

Auch sollte man erste sich ein Steckbrett zum erstmaligen anschließen anschaffen und um die Funktion zu testen.

Als Gehäuse kann man aus dem Baumarkt eine fertige Holzkiste verwenden in der man Löcher für die Schalt reinbohrt und ggf. eine zweite Ebene usw.

Kosten liegen bei 250€.

Funktionsumfang der Jukebox

Die Jukebox kann mit Hilfen der RFID Karten, welche vor dem Lesegerät gehalten werden, die Musik abspielen, welche sich auf dem Pi befinden. Gleichzeitig kann man den Player auch über ein Webinterface steuern. Hierzu muss man nur im selben Netzwerk wie der Pi sein und auch seine IP-Adresse kennen. Die Taster dienen um Standardfunktionen zu bedienen. (Lauter, Leiser, nächster Track, Track zurück, Play/Pause, Ein-/Ausschalten, Reset) Es wird zuvor hinter jeder RFID Karte verschiedene Ordner mit Musik oder Streams hinterlegt. Diese werden der RFID Karte zugeordnet und dann wiedergegeben.

1. Installieren von Rasbian

Als Image habe ich 2017-02-16-raspbian-jessie verwendet.

Mit Win32DiskImager kann man das Image einfach auf die SD-Karte überspielen. (Image auswählen und den Laufwerksbuchstaben der SD-Karte wählen und auf „write“ drücken.

Jetzt SD Karte in den Pi stecken. Maus, Tastatur, Netzteil und Bildschirm anschließen und ggf. Schalter vom Netzteil aktivieren.

Nach der Installation booten lassen.

Jetzt SSH aktivieren.

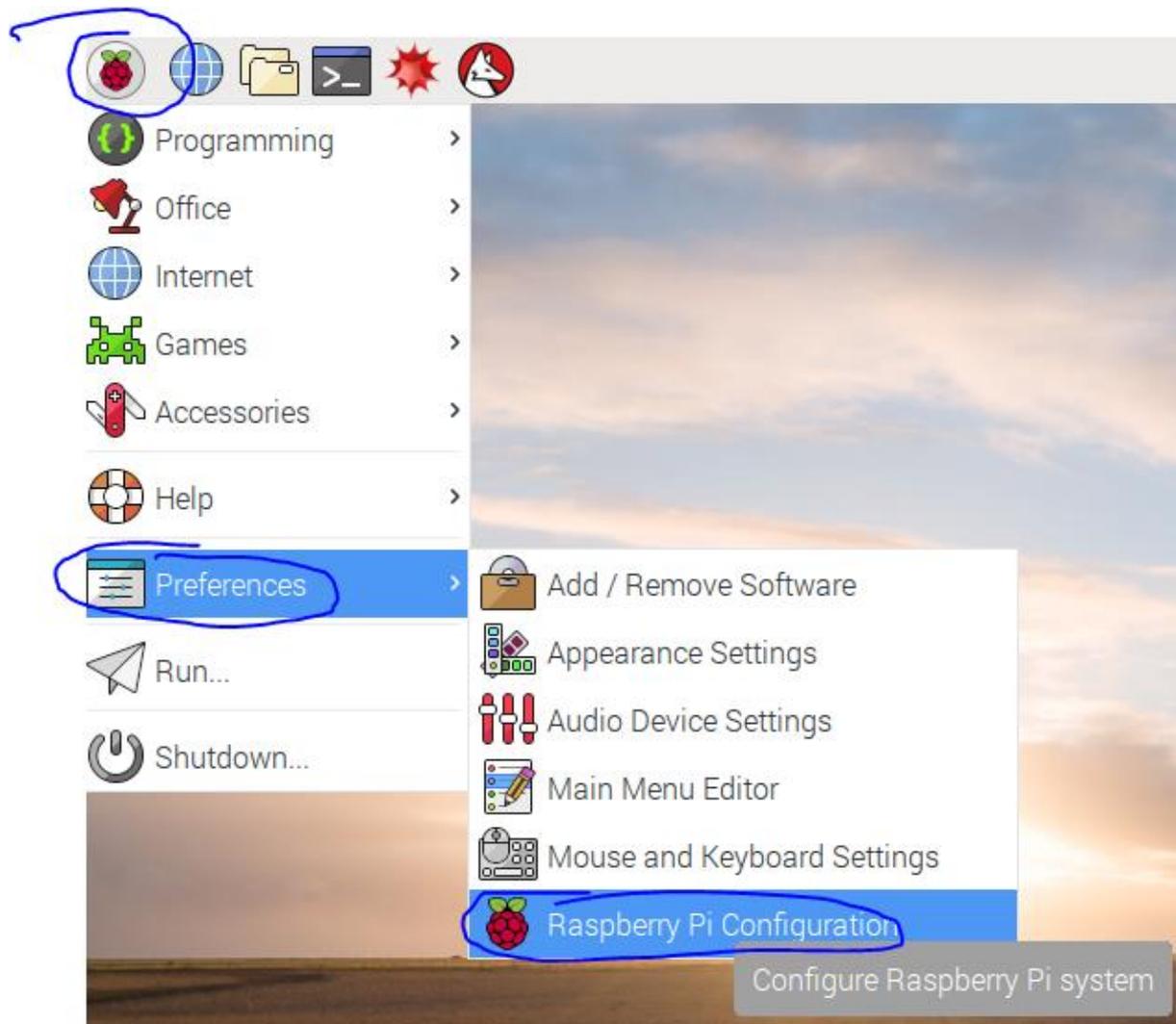


Bild von <https://raspberrypi-tutorials/ssh-am-raspberry-pi-aktivieren-neue-methode-mit-raspbian-25-11-2016/>

Hier kann man unter den weiteren Einstellungen die Sprache und Tastatur auf Deutsch einstellen. So wie das Passwort, welches man erste nach den Tastatureinstellungen ändern sollte, aktuell raspberry, ändern. Sollte selbsterklärend sein. Wichtig ist aber SSH zu aktivieren. Später kann man

dem Pi hier auch sagen, dass er nicht mehr in den Desktop-Modus starten soll damit das Booten schneller geht. Oben Rechts bei der Uhr kann man auch schon die WLAN-Einstellungen vornehmen.

2. Einloggen über SSH am PC mit dem Programm Putty

Hierzu IP Adresse herausfinden des Raspberry Pi z.B. im Fritzbox Menü.

Einloggen mit dem User pi und Passwort raspberry. (oder dem zuvor neu gewählten)

3. Erstmal führen wir Updates/Upgrades aus

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

Dies dauert je nach Internetanbindung etwas.

4. **Python installieren** (Hinweis das Kursiv gedruckte kann durch das einfache kopieren der Zeile in die Zwischenablage kopiert werden und durch die rechte Maustaste in Putty eingefügt werden)

Befehl

```
sudo apt-get install python-dev python-pip gcc
```

jetzt Version herausfinden

```
uname -r
```

Ergebnis in etwa so aus 4.4.48-v7+

Brauchen wir bei der weiteren vorgehensweise

```
sudo apt-get install linux-headers-4.4
```

evdev Code installieren

```
sudo pip install evdev
```

5. PHP Installieren

```
sudo apt-get install lighttpd php5-common php5-cgi php5
```

Konfigurieren

```
sudo nano /etc/lighttpd/lighttpd.conf
```

Hier die Zeile wie folgt anpassen:

```
server.document-root = "/home/pi/RPi-Jukebox-RFID/htdocs"
```

Nun geben wir dem PHP Skript Root Rechte

```
sudo nano /etc/sudoers
```

ganz unten tragen wir folgendes ein:

```
www-data ALL=(ALL) NOPASSWD: ALL
```

Nun aktivieren ...fastcgi

```
sudo lighty-enable-mod fastcgi-php
```

Jetzt Webserver neustarten mit

```
sudo service lighttpd force-reload
```

6. VLC installieren

```
sudo apt-get install vlc
```

Danach wird VLC als root per webserver bedient.

```
sudo sed -i 's/geteuid/getppid/' /usr/bin/vlc
```

7. Installieren mpg123

```
sudo apt-get install mpg123
```

8. Installieren von git

```
sudo apt-get install git
```

9. Jukebox von Micz Flor installieren

Hierzu ins home Verzeichnis wechseln.

```
cd /home/pi/
```

```
git clone https://github.com/MiczFlor/RPi-Jukebox-RFID.git
```

10. Installieren von SAMBA, damit man mit dem PC über die Netzwerkfreigabe auf den Pi zugreifen kann.

```
sudo apt-get install samba samba-common smbclient
```

Danach prüfen ob SAMBA läuft

```
sudo service smbd status
```

```
sudo service nmbd status
```

Ergebnis sollte active running sein.

Nun erstellen wir ein Verzeichnis mit Benutzerrechten

```
sudo chmod 770 /home/pi/RPi-Jukebox-RFID/shared/
```

Neue Grundkonfiguration von SAMBA, alte Datei verschieben.

```
sudo mv /etc/samba/smb.conf /etc/samba/smb.conf_alt
```

Neue anlegen

```
sudo nano /etc/samba/smb.conf
```

Jetzt folgendes einfügen

```
[global]
```

```
workgroup = WORKGROUP
```

```
security = user
```

```
encrypt passwords = yes
```

```
[pi_jukebox]
```

```
comment= Pi Jukebox
```

```
path=/home/pi/RPi-Jukebox-RFID/shared
```

```
browseable=Yes
```

```
writable=Yes
```

```
only guest=no
```

```
create mask=0777
```

```
directory mask=0777
```

```
public=no
```

Nun speichern und verlassen

STRG+X "j" "ENTER"

Jetzt testen mit

```
testparm
```

Jetzt SAMBA neu starten mit diesen Befehlen

```
sudo service smbd restart
```

```
sudo service nmbd restart
```

Nun muss noch ein Passwort hinterlegt werden. Dazu den Befehl eingeben.

```
sudo smbpasswd -a pi
```

Am besten nimmt man das gleiche wie für den Benutzer pi.

(raspberry)

Dann den Benutzer Pi deaktivieren und wieder aktivieren

```
sudo smbpasswd -d pi
```

```
sudo smbpasswd -e pi
```

Die Freigabe kann am PC geprüft werden durch folgende Eingabe im Arbeitsplatz

[\\IP-Adresse](#)desPis

Dann Benutzername und Passwort eingeben.

11. Installation HifiBerry miniAmp und ALSA (wer die Soundkarte vom Pi nutzt mit externen Lautsprechern braucht dies nicht)

ALSA installieren und konfigurieren

```
sudo apt-get install alsa-utils
```

Standard Zeile löschen

```
sudo nano /boot/config.txt
```

Jetzt nach dtparam=audio=on suchen und mit # unscharf schalten

Nun folgende Zeile hinzufügen:

```
dtoverlay=hifiberry-dac
```

Jetzt nano mit STRG+X beenden. Datei aber speichern mit „J“.

HifiBerry / ALSA Anpassungen

Datei öffnen

```
sudo nano /etc/asound.conf
```

Folgendes eintragen:

```
pcm.hifiberry {  
    type softvol  
    slave.pcm "plughw:0"  
    control.name "Master"  
    control.card 0  
}
```

```
pcm.!default {  
    type plug  
    slave.pcm "hifiberry"
```

}

Jetzt nano mit STRG+X beenden. Datei mit „j“ speichern.

sudo reboot

12. Installation RFID Leser

Gerät per USB anschließen. Es sollte dann ein „beep“ kommen. Dieser wird immer wieder kommen beim Einstecken und auch wenn ein Chip über das Gerät gezogen wird.

Jetzt muss die LED Lampe am Reader aufleuchten und wenn ein Chip oder eine Karte davor gehalten wird muss es „beep“ machen.

Jetzt prüfen wir ob der Reader auch am Pi reagiert.

ls -la /dev/input/by-id/

Ganz oben das USB-Gerät mit event 3 ist mein leser “event3 reagiert”

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Oct 26 13:34:36 2017
pi@jukebox:~ $ ls -la /dev/input/by-id/
insgesamt 0
drwxr-xr-x 2 root root 140 Okt 26 13:57 .
drwxr-xr-x 4 root root 200 Okt 26 13:57 ..
lrwxrwxrwx 1 root root  9 Okt 26 13:57 usb-HXGCoLtd_HIDKeys-event-kbd -> ../event3
lrwxrwxrwx 1 root root  9 Okt 26 13:34 usb-Logitech_USB_Keyboard-event-if01 ->
../event1
lrwxrwxrwx 1 root root  9 Okt 26 13:34 usb-Logitech_USB_Keyboard-event-kbd ->
../event0
lrwxrwxrwx 1 root root  9 Okt 26 13:34 usb-Logitech_USB_Optical_Mouse-event-mouse ->
../event2
lrwxrwxrwx 1 root root  9 Okt 26 13:34 usb-Logitech_USB_Optical_Mouse-mouse ->
../mouse0
pi@jukebox:~ $
```

Registrierung des Readers für die Jukebox

cd /home/pi/RPi-Jukebox-RFID/scripts/

python2 RegisterDevice.py

Jetzt den Barcodescanner wählen den wir zukünftig nutzen wollen.

Bei mir 0

Choose the reader from list

0 HXGCoLtd HIDKeys

1 Logitech USB Optical Mouse

2 Logitech USB Keyboard

3 Logitech USB Keyboard

Device Number: 0

Um zu prüfen ob der Reader gewählt wurde folgendes eingeben:

```
cat deviceName.txt
```

Jetzt sollte eine Zeile kommen wo am Anfang der Name des Readers kommt.

```
HXGCoLtd HIDKeys
```

13. AutoStart Jukebox per Cron

Cron den Befehl zum Autostart geben

Dazu im putty folgendes eingeben:

```
crontab -e
```

Jetzt wählen wir nano, da dieser uns schon bekannt ist und tragen am Ende folgendes ein.

```
@reboot mpg123 /home/pi/RPi-Jukebox-RFID/misc/startupsound.mp3
```

```
@reboot python2 /home/pi/RPi-Jukebox-RFID/scripts/daemon_rfid_reader.py &
```

Jetzt wieder schließen und speichern STRG+X, j, Enter

14. Kopieren des Media Player Skript

Wir gehen in das Verzeichnis

```
cd /home/pi/RPi-Jukebox-RFID/scripts/
```

Kopierbefehl

```
cp rfid_trigger_play.sh.sample rfid_trigger_play.sh
```

```
chmod +x rfid_trigger_play.sh
```

Rebooten

```
sudo reboot
```

15. RFID Karten oder Chips registrieren

Nach dem Hochfahren ziehen wir eine Karte über das Lesegerät.

Jetzt gehen wir über SMB auf den Pi also in das Netzwerkverzeichnis shared, welches wir freigegeben haben.

Einfach in der Adresszeile im Arbeitsplatz oder Computer [\\IP-AdressedesPis](#) eingeben.

In der Datei latestID.txt steht nun was das für eine Karte ist (die ID) und auch ob die Karte schon zuvor verwendet wurde.

Beispielnachricht

```
Card ID '0003775476' was used at '2017-10-26.17:13:29'.
```

This ID was used for the first time.

The shortcut points to audiofolder '0003775476'.

Damit man später nicht nur Ordner mit den IDs hat sondern, damit man direkt weiß um was für Audiofiles es geht kann man die ID ersetzen gegen richtige Namen.

Um dies durchzuführen öffnen wir den Ordner shortcuts auf dem pi. Hierin befindet sich nun eine Datei mit dem Namen der RFID Nummer (Id). Diese Datei öffnen wir mit Proton einem Texteditor.

Jetzt können wir die ID umbenennen in einen Namen. Z.B. test. Danach drücken wir wieder speichern oben am Rand.

Jetzt müssen wir das Verzeichnis erstellen worin die Musikdateien der RFID Karte „test“ sind.

Und dann fügen wir die Musikdateien ein die zu dem Archiv gehören. Wichtig ist das diese immer nach Alphabet abgespielt werden. Also bei Hörspielen auf die Bezeichnung achten.

Um einen Webstream hinzuzufügen gehen wir genauso vor.

Als Datei im RFID Ordner erstellen wir aber eine .txt Datei mit dem Editor von Windows und tragen dort in der ersten Zeile die Linkadresse des Streams ein.

16. HifiBerry Einstellungen im VLC (wer die Soundkarte vom Pi nutzt mit externen Lautsprechern braucht dies nicht)

Unter Einstellungen-Audio-Output Module

ALSA Audioausgabe

Bei Gerät

Hifiberry wählen

Hier kann auch die maximale Lautstärke von VLC gewählt werden. Ich habe damit die Lautsprecher nicht übersteuern hier 100% angegeben.

17. Buttons einbinden

gpiozero Python 3

```
sudo apt install python3-gpiozero
```

gpiozero Python 2

```
sudo apt install python-gpiozero
```

Die Datei zerobutton.py in das Homeverzeichnis auf dem Pi einfügen.

Jetzt muss die Datei noch ausführbare Rechte bekommen. Dazu über putty ins Homeverzeichnis und folgendes eingeben.

```
sudo chmod a+x zerobutton.py
```

Danach muss die Datei in den Autostart geschoben werden bzw. am Start ausgeführt werden.

```
sudo nano /etc/rc.local
```

und wir tragen ganz unten vor exit

```
/home/pi/zerobutton.py &
```

18. Anpassungen für Webinterface (wer die Soundkarte vom Pi nutzt mit externen Lautsprechern braucht dies nicht)

Da wir nicht die normale Soundkarte des Pis nutzen müssen wir im Skript von Micz Flor ein paar Dinge anpassen, sonst kann man im Webinterface nicht die Lautstärke anpassen.

```
cd /home/pi/RPi-Jukebox-RFID/htdocs
```

```
sudo nano index.php
```

Hier folgende Zeile anpassen:

```
// change volume
```

```
if(isset($_urlparams['volume'])) {
```

```
    exec("/usr/bin/sudo amixer sset Master ".$_urlparams['volume']."%");
```

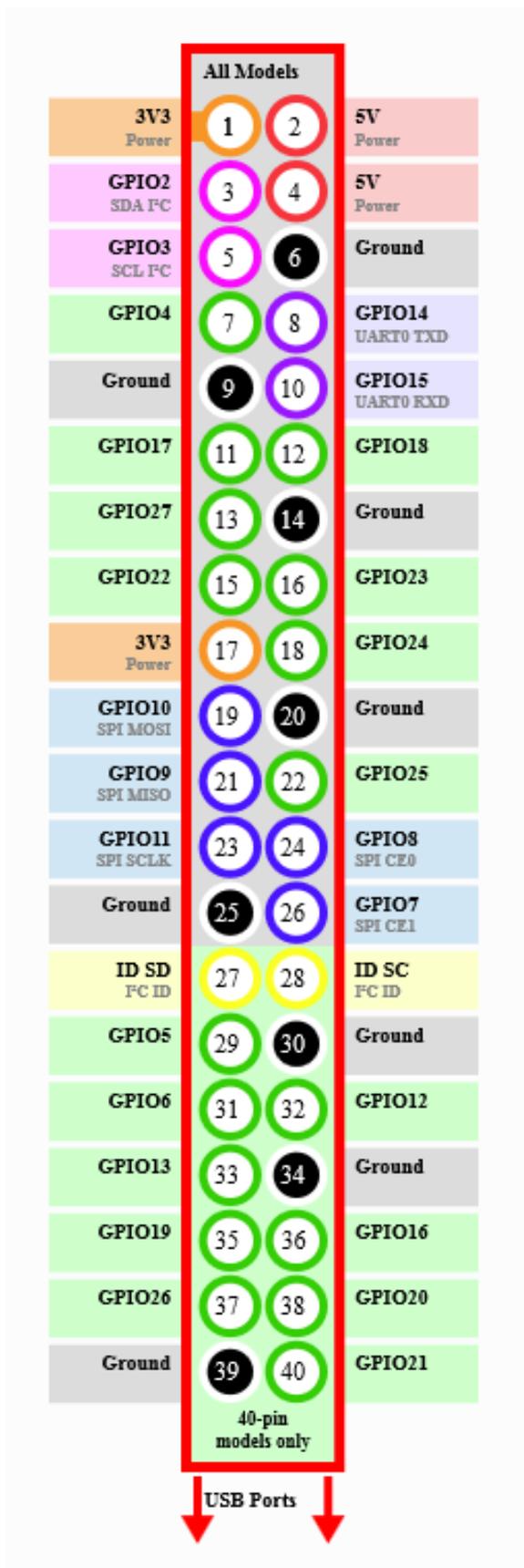
```
    /* redirect to drop all the url parameters */
```

```
    header("Location: ".$_conf['url_abs']);
```

```
    exit;
```

!!!Wichtig ist die Lücke nach Master!!!

19. Pin-Belegung am Pi

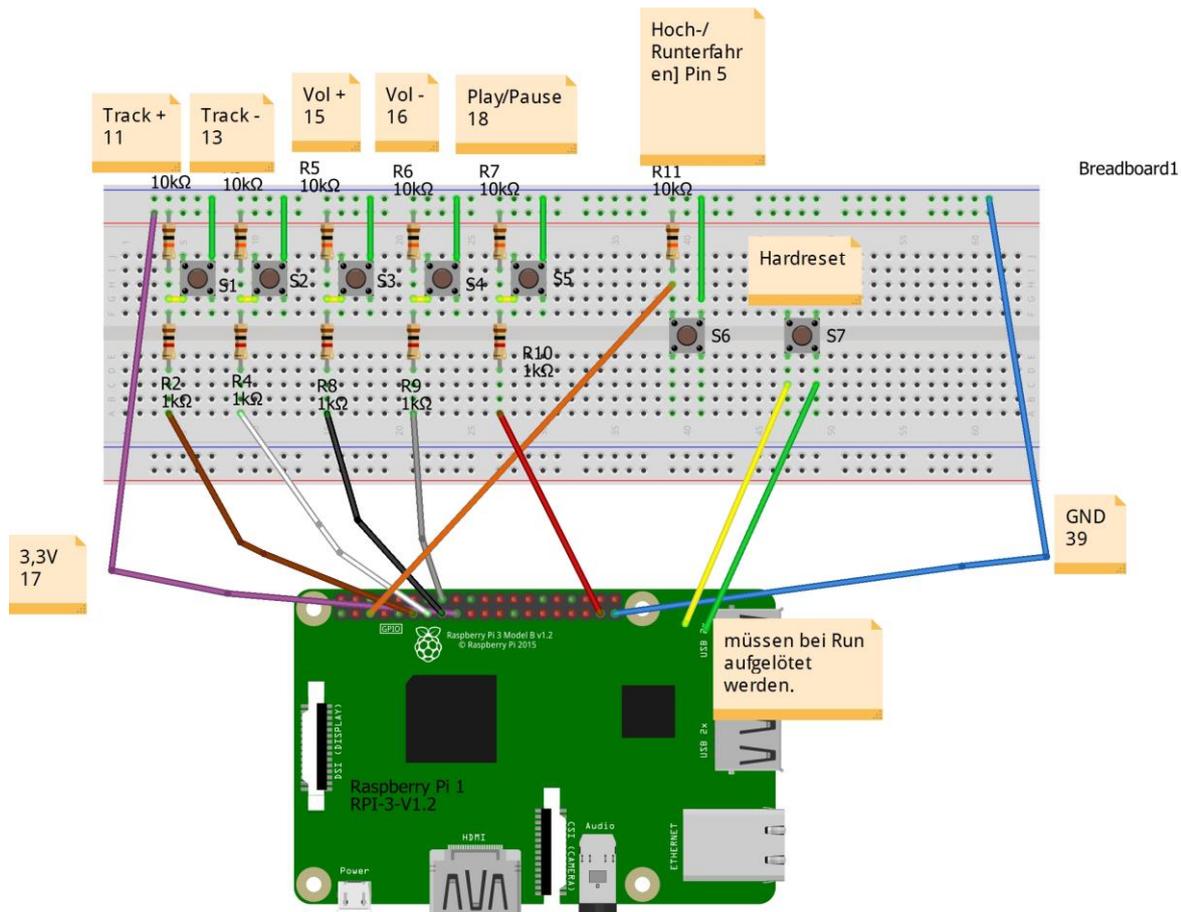


1,3,5,19,21,23,27,35,4,6,12,24,26,28,38,40 sind belegt durch den HifiBerry.

Meine Buttons habe ich an:

Pin	GPIO	Funktion
5	3	Ein-Ausschalter (Doppelbelgung)
15	22	Vol +
16	23	Vol -
11	17	Track +
13	27	Track -
37	26	Play/Pause

Bemerkung: Pin 5 ist vom Berry und vom Schalter belegt. Das Problem mit Ein- / Ausfunktion ist, dass diese nur über diesen Pin funktioniert. Bzw. kann man das ordentliche Herunterfahren auch an einem anderen Pin machen aber dann fährt der Pi nicht mehr hoch.



fritzing

Hier ist der HifiBerry nicht abgebildet. Anschlüsse sind Pins.

Um die Verdrahtung zu verdeutlichen versuche ich es nun zu beschreiben. Am Ende der Anleitung sind auch noch Bilder der einzelnen Platinen.

Ich habe sämtliche Pins, welche ich benötige vom Pi auf zwei extra Platinen geführt. Die eine Platine ist nur für den HifiBerry, damit ich diesen an seiner Schnittstelle direkt aufsetzen kann. Die zweite Platine ist wie oben zu sehen für die Taster. Es geht lediglich ein Kabel von der HifiBerry Platine zu der Tasterplatine. Zusätzlich falls der Pi mal abstürzt habe ich eine kleine dritte Platine erstellt für einen Hardreset. Diese Pins müssen aber auf den Pi auch erste aufgelötet werden.

20. Programmierung

Mein Dank geht an das Forum www.forum-raspberrypi.de. (hyle, Assarbad, linusg, Geliras, tullm, MarkBinary und natürlich Micz Flor, welcher den kompletten Skript geschrieben hat für das Webinterface usw.) Wenn ich Jemanden vergessen haben, dann tut es mir leid. Hierfür einfach die Datei benutzen aus dem Anhang. (zerobutton.py)

```
#!/usr/bin/python3
```

```
# Datei für die Taster
```

```
from gpiozero import Button
```

```
from signal import pause
```

```
from subprocess import check_call
```

```
def def_shutdown():
```

```
    check_call(['sudo', 'poweroff'])
```

```
def def_volU():
```

```
    check_call("amixer sset Master 10+", shell=True)
```

```
def def_volD():
```

```
    check_call("amixer sset Master 10-", shell=True)
```

```
def def_next():
```

```
    check_call("echo 'next' | nc.openbsd -w 1 localhost 4212", shell=True)
```

```
def def_prev():
```

```
    check_call("echo 'prev' | nc.openbsd -w 1 localhost 4212", shell=True)
```

```
def def_halt():
```

```
    check_call("echo 'pause' | nc.openbsd -w 1 localhost 4212", shell=True)
```

```
shut = Button(3, hold_time=2)
```

```
volU = Button(22)
```

```
volD = Button(23)
```

```
next = Button(17)
```

```
prev = Button(27)
```

```
halt = Button(26)
```

```
shut.when_held = def_shutdown
```

```
volU.when_pressed = def_volU
```

```
volD.when_pressed = def_volD
```

```
next.when_pressed = def_next
```

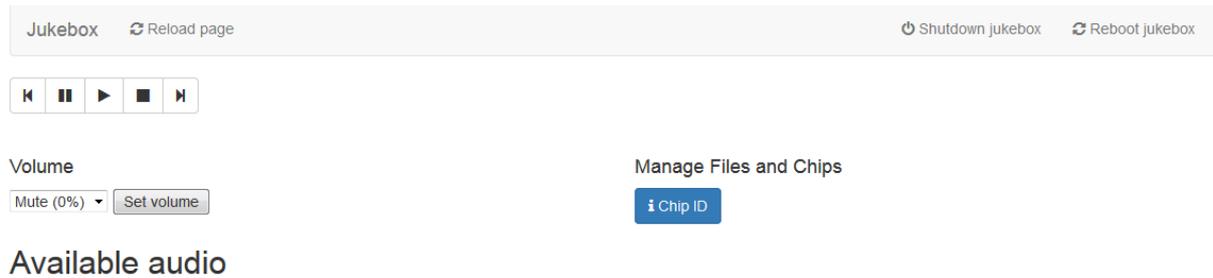
```
prev.when_pressed = def_prev
```

```
halt.when_pressed = def_halt
```

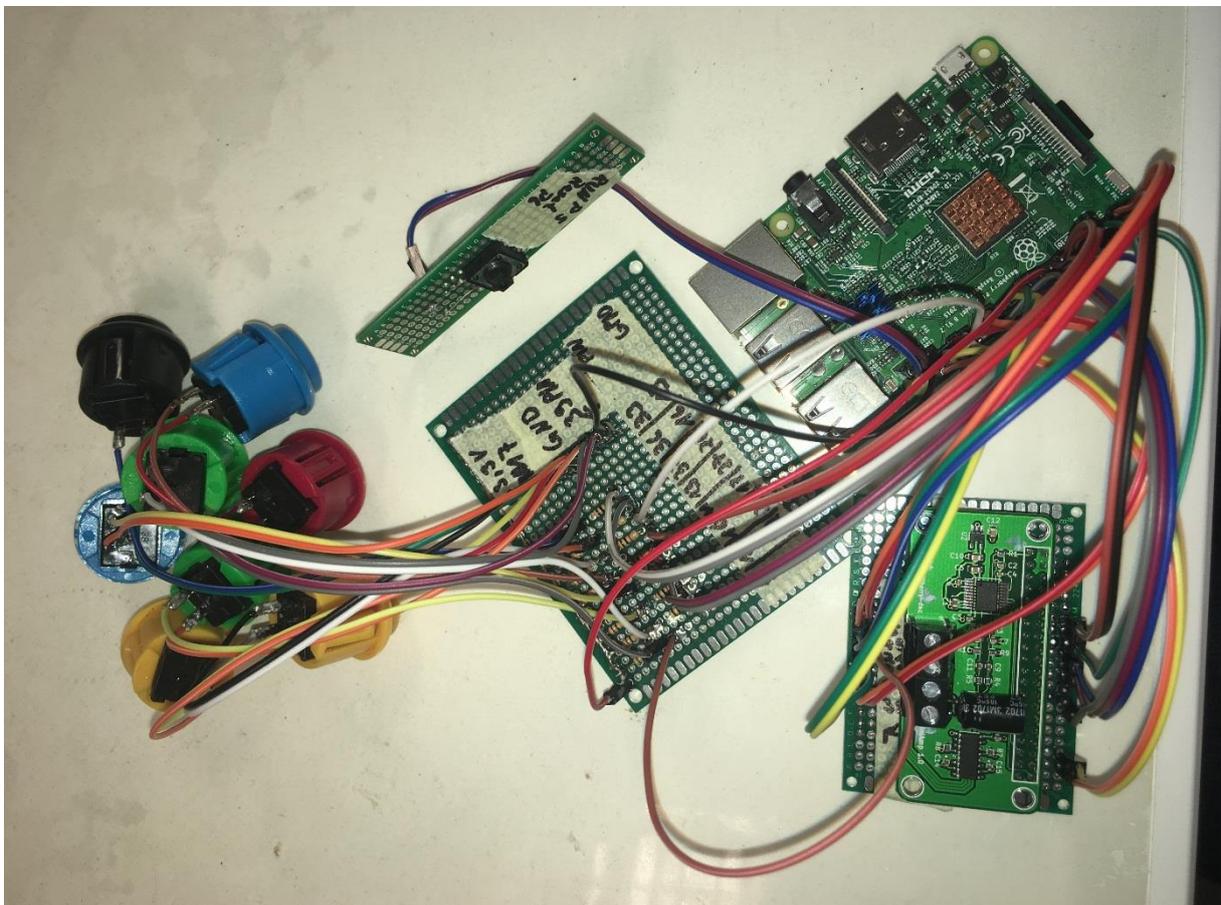
```
pause()
```

21. Jukebox per WebAPP bedienen.

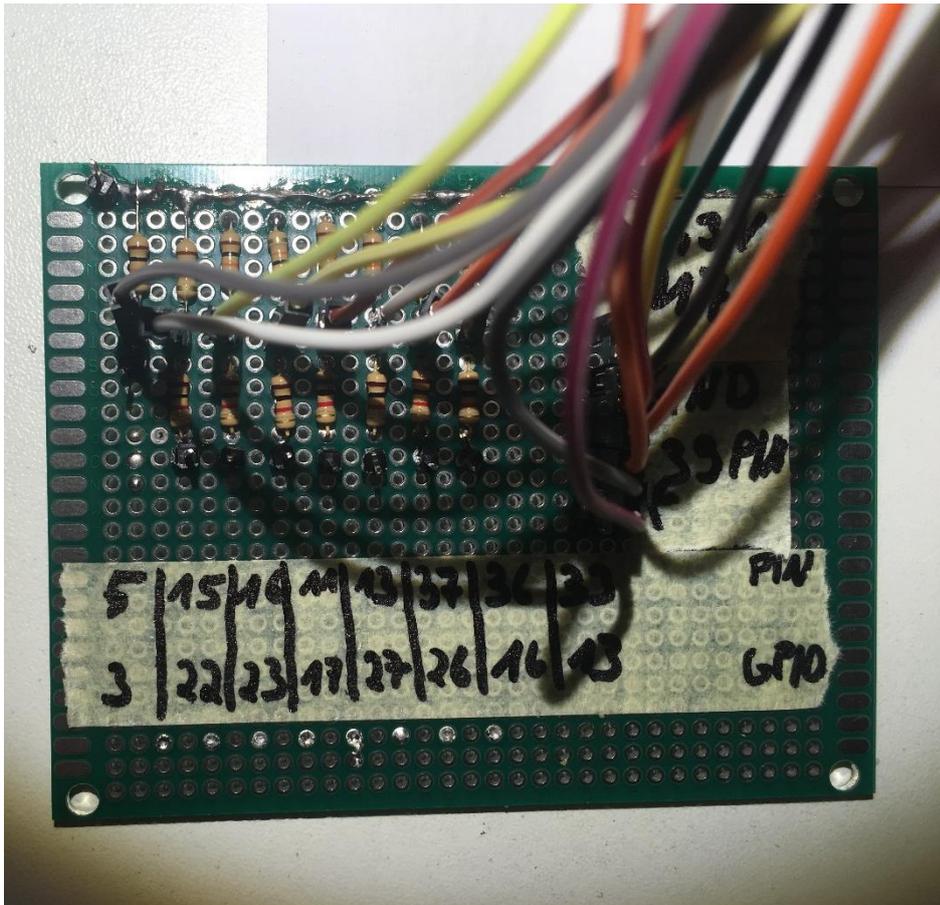
Hierzu einfach die IP Adresse des Pis eingeben. Danach seid ihr im Bedienmenü.



22. Bilder der gelöteten Platinen

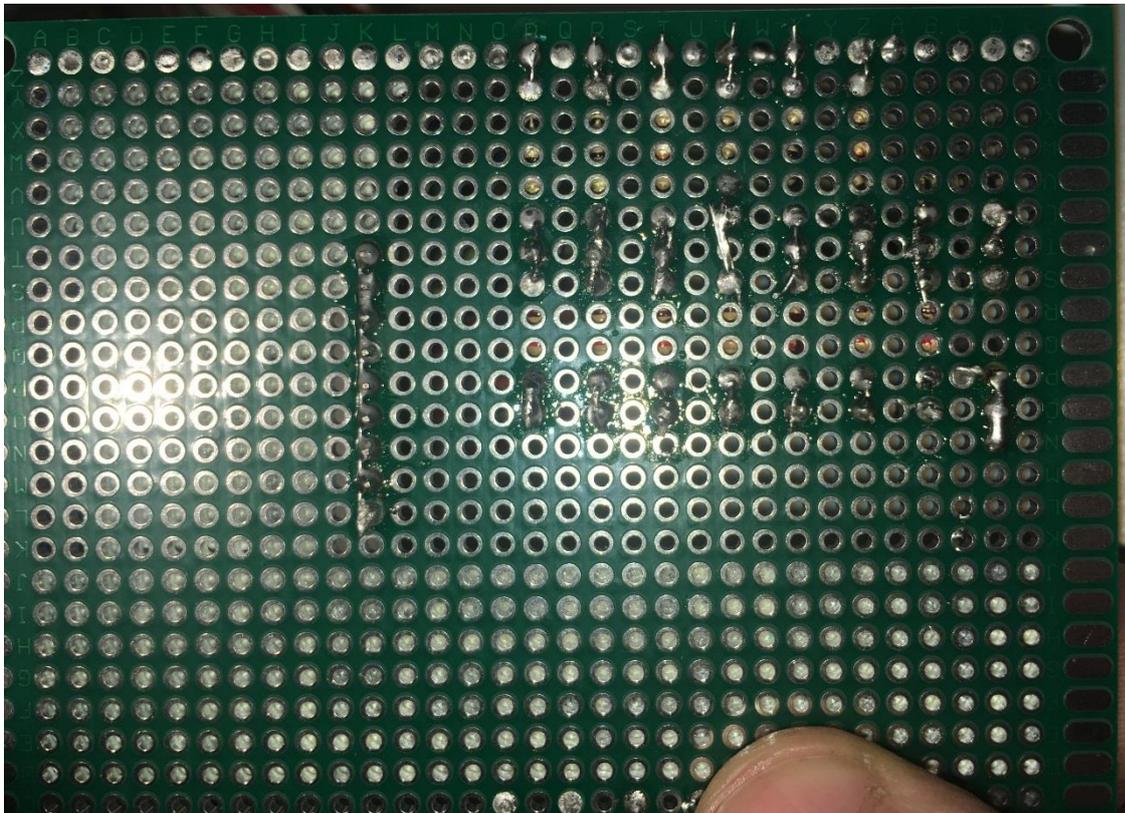


Platinen mit Verkabelung

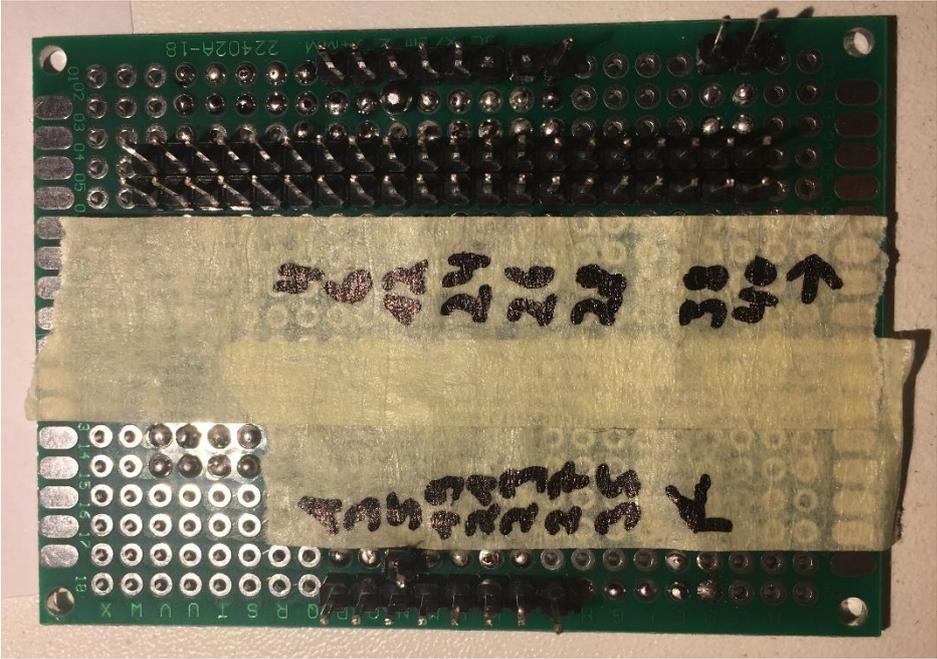


Schalter-Platine Vorderseite (Oben 3,3V von Pin 17, Rechts GRND von Pin 39)

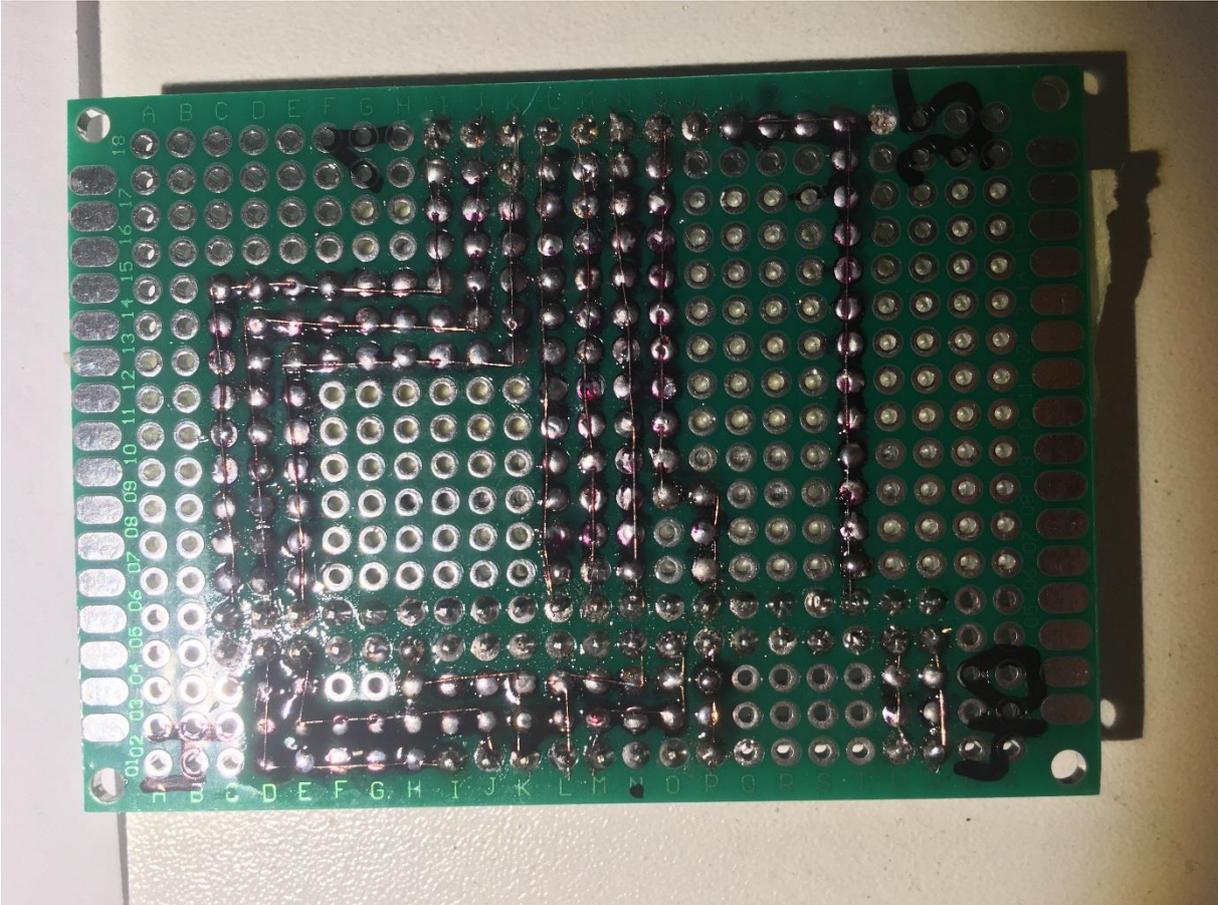
Pin 5 wird mit Pin 5 auf der Hifiberry Platine verbunden.



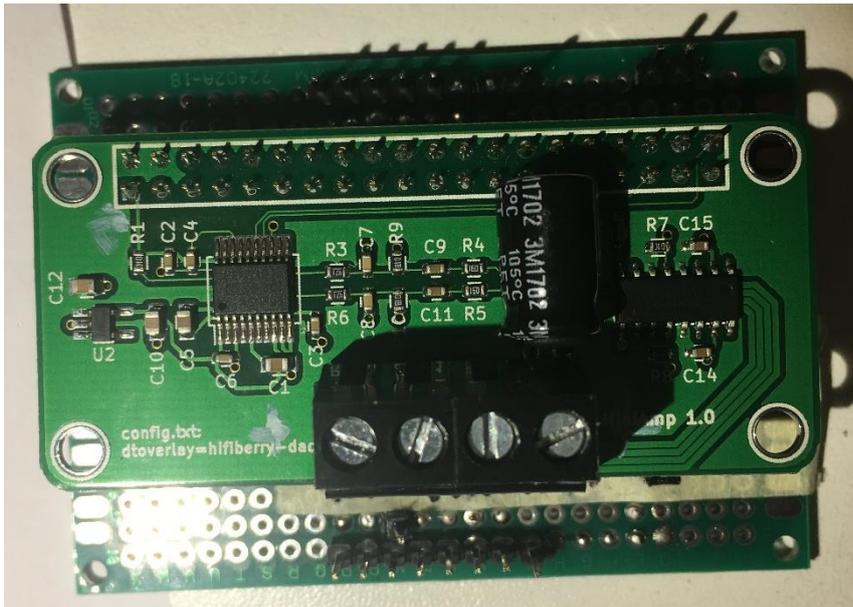
Rückseite der Schalter-Platine (Achtung hier sind zwei Kontakte nicht richtig verbunden)



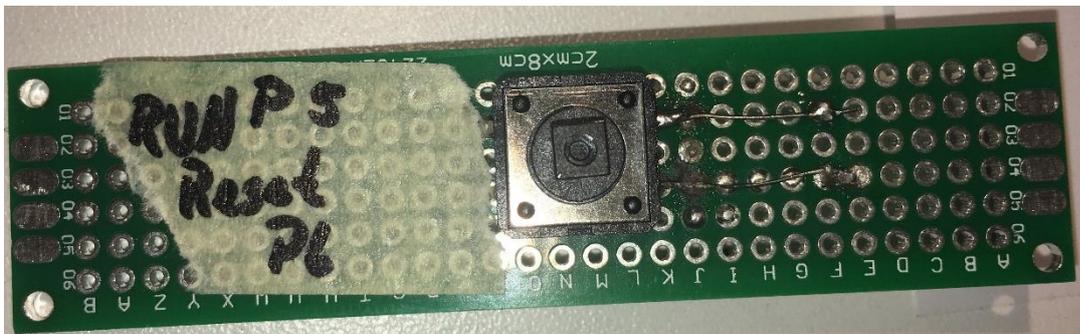
Aufsteckplatine für HifiBerry



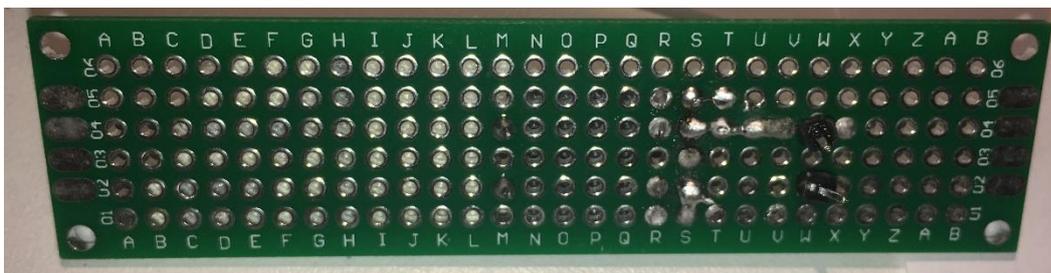
Rückseite Aufsteck-Platine HifiBerry



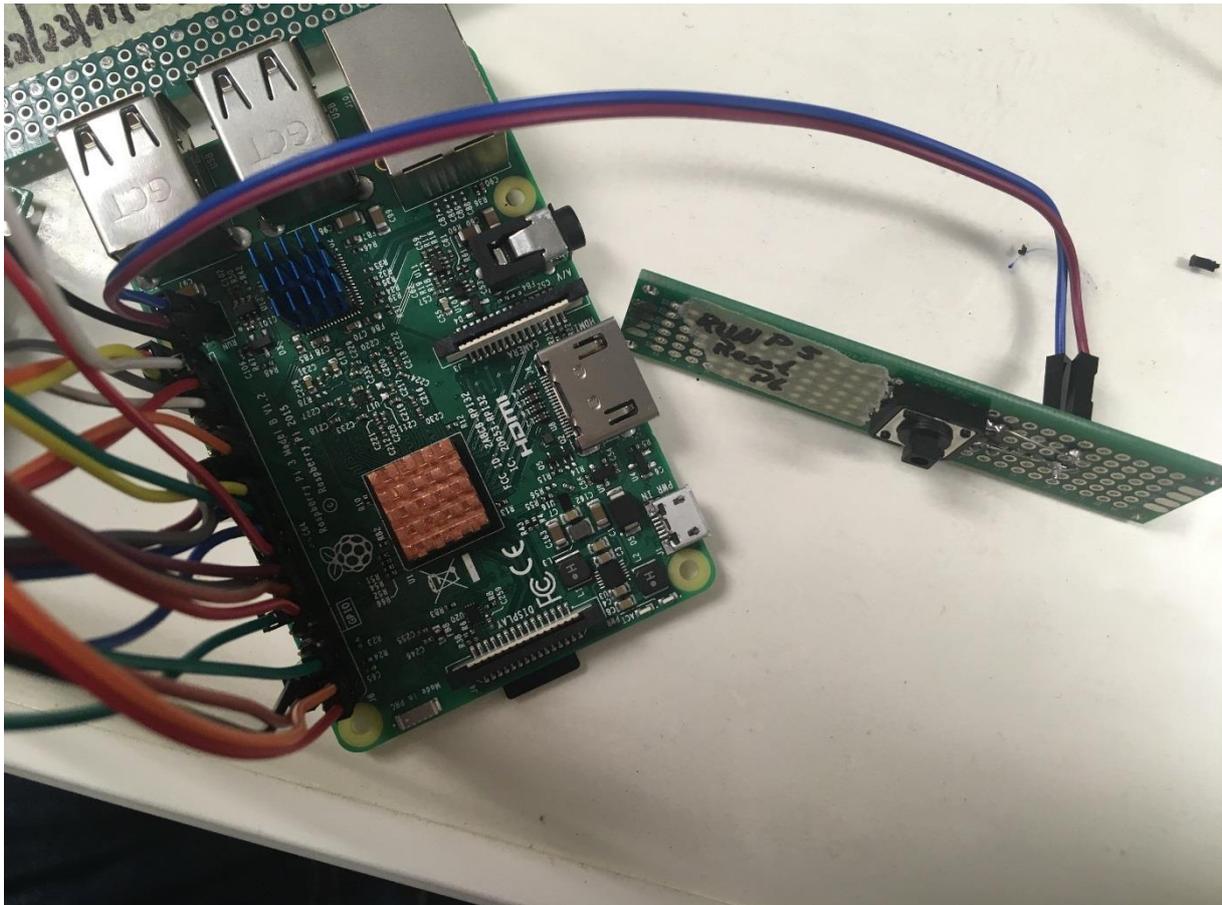
Aufsteckplatine mit HifiBerry



Reset Platine Vorderseite



Rückseite Reset-Platine Die Pins werden direkt auf die zusätzlichen Pins am Pi gesteckt



Zusätzliche Pins für Hardreset am Pi.

23. Danksagung

Mein Dank geht an das Forum www.forum-raspberrypi.de. (hyle, Assarbad, linusg, Geliras, tullm, MarkBinary und natürlich Micz Flor, welcher das komplette Skript geschrieben hat für das Webinterface usw.) Wenn ich Jemanden vergessen haben, dann tut es mir leid.

Forumseinträge

<https://forum-raspberrypi.de/forum/thread/13144-projekt-jukebox4kids-jukebox-fuer-kinder/?pageNo=8>

Micz Flor auf Git mit Anleitung

<https://github.com/MiczFlor/RPi-Jukebox-RFID>