

**Vortrag für das Linuxcafe am 05.10.2015**

## **Bau eines WiFi Radios mit Raspberry Pi B+**

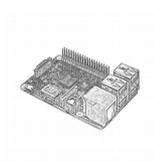
### **Ursprüngliche Idee:**

Bau eines WiFi Radios für die Küche mit gutem Sound, ohne Knöpfe. Bedienung sollte über Touchdisplay erfolgen und die Auswahl der Sender über den Streamer Radiotray.

Der Gedanke dieses selber zu bauen war zum Einen aus Interesse. Der hauptsächliche Grund war, dass die im Verkauf angebotenen Internetradios sich in einer Preisklasse von 100€ aufwärts bewegen. Der geschätzte Durchschnittspreis bei >147€. Bei den finanziell erschwinglichen Radios, konnte ich mir aber nicht vorstellen, dass diese physikalisch bedingt einen vernünftigen Sound hervorbringen. Daneben sind die meisten Internetradios mit Diensten im Internet verknüpft (Spotify, Apps, etc). Man kann gekaufte Radios schlecht reparieren und hat nicht die volle Kontrolle über Hardwar- und Software.

Die Hardware welche ich mir gekauft hatte, waren so ausgewählt, dass diese auch anderweitig nutzbar sind, falls ich meines Radios überdrüssig sein sollte.

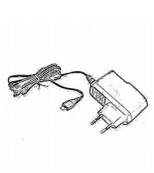
**Hardware:** Gesamtkosten 157€  
(gerundete Preise)



Raspberry Pi B+ 30€



300 Mbits/s USB WLAN Dongle 13€



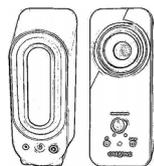
5V 2000mAH Ladegerät 10€



16GB Class 10 SDHS 9€



HiRes USB-DAC SABRE 24/96 mit ES9023 50€



Creative Inspire T12 2.0 Lautsprecher 45€

## **Durchführung:**

### Probleme:

Das Einrichten des Rasbian an sich war kein Problem, zumal die Installation gut dokumentiert ist. Probleme gab es mit dem Sound. Out of the box kam aus den Lautsprechern kein Ton heraus, obwohl alle Komponenten erkannt wurden. Es war nur unter größerem Aufwand möglich, Alsa dazu zu bewegen Töne heraus zu spucken, bzw. ging es nur über Umwege mit Pulse Audio. Es wären viele Konfigurationen nötig gewesen um den sound zu optimieren (Impedanzen, Bufferraten, Frequenzen usw.). Hierzu hat es mir an Zeit und technischem know how gefehlt. Die Ursache war das Zusammenspiel der Kernelmodule mit dem ES9023 ein 24-bit-sterio DAC. Nach vielen probieren hat es funktioniert, aber mit schlechter Qualität. Man muss ja das Rad nicht neu erfinden habe ich mir gedacht und verschiedene Hifiplayer getestet. Zu nennen sind da: piCore Player, Pi MusicBox, rune Audio, Volumio und SqueezePlug.

Das meiste hat mich nicht wirklich begeistert. Entweder war die Bedienung nicht komfortabel, bzw. die Bedienoberfläche zu unübersichtlich, oder die Reaktionszeiten der player waren zu lange. Daneben waren die player zum Teil eingeschränkt, was die Abspielformate (mp3, AAC,Ogg Vorbis) angeht. Es konnten aber auch nicht alle Playlist formate (m2u, pls, url) benutzt werden. Die streamingfähigkeit (Schnelligkeit, Konsistenz) hat sich auch sehr unterschieden.

### Lösung:

Entschieden habe ich mich für eine Kombination, basierend auf Squeeze Box technologie, deren Grundlage der Logitechserver ist. Die Jungs von SqueezePlug (squeezebox, wäre aber auch über airplay möglich) und Max2play (shop, framework, plug ins) haben großes geleistet. Hut ab! Es gibt eine kostenlose Version, in der nicht alle Erweiterungen freischaltbar sind. Und eine Version für 10€. Die kostenlose Version funktioniert bestens. Zu dem sind weiter Features möglich, wie Kodi, Accses point, Multiroom und homeautomatic usw.. Die Einrichtung ist einfach und sehr gut, zum Teil mit Videos, dokumentiert. Vor allem auch für Nicht Linux Kenner.

Anbindungen zum NAS, Samba freigaben, etc. alles einfach konfigurierbar. Bedienen kann man dies über eine browserbasierte Administrationsoberfläche. Also platform unabhängig. Vor allem kann man headless installieren und einrichten. Die Konfiguration des Raspberries war sehr einfach mit super sound. Die Bedienung ist von allen Playern am schnellsten und es sind alle Formate möglich.

So sieht jetzt mein Raspberry aus:

Das Raspberry und die Aktivboxen sind über eine Zweifachsteckdose mit Ausschalter verbunden. Beim Einschalten der selbigen, fährt das Raspberry hoch und der Logitechserver startet. Mit einem Tablet, Handy oder Telefon starte ich squeezcontroller. Das ist eine APP mit der ich auf den Logitechserver zugreife. Also die Auswahl, was ich hören will, Radio, USB, oder Poadcasts oder.... Natürlich ist auch eine Steuerung über Pc möglich (Broswer). Den Raspberry fahre ich über den Broswer (Aufruf über IP des Raspberry auf der Administrationsoberfläche) runter. Nach dem der Raspberry herunter gefahren ist, kann man die Stromzufuhr kappen (Raspberry ist empfindlich bei unsachgemäßen beenden). Da die Software hinter dem Router und einer Firewall sitzt, ist das ganze auch relativ sicher. Man könnte noch zusätzliche Passwörter vergeben. Der Raspberry selber ist an einem Bild befestigt.

**Was noch geplant ist:**

Ein Script schreiben, damit wenn das Raspberry hochfährt, mein Lieblingsradiosender automatisch startet.

Ein Gehäuse für das Raspberry.

Eventuell mit Anbringung des Touchscreens, welches ich ursprünglich gekauft hatte, soll sogar demnächst von Max2play unterstützt werden.

**Resüme:**

Zum Teil bin ich etwas blauäugig an die Sache heran gegangen und habe mit weniger Schwierigkeiten gerechnet. Ich hatte zwar bei der Hardware lange gesucht und auf Kompatilität zum Raspberry geachtet. Aber die Probleme mit dem Dac hatte ich nicht erwartet.

Ich würde das ganze aber jeder Zeit wieder machen. Der Sound ist jetzt wirklich gut, das Radio nimmt nicht viel Platz weg (vorher hatten wir ein Röhrenradio) und das die Bedienung mit Tablet so komfortabel ist, hätte ich nicht gedacht.

## Nützliche Links zu dem Bau eines WiFi Radios mit dem Raspberry

### Hardware:

[Internetradios](#)

[Raspberry B+](#)

[Netzteil](#)

[WiFi Dongle](#)

[DAC](#)

[Lautsprecherboxen](#)

### Datenblätter

[Data Raspberry](#)

[Data DAC](#)

[Data Boxen](#)

### Software, Player

[volumio](#)

[runeaudio](#)

[pimusicbox](#)

[squeezeplug](#)

[mysqueezebox](#)

[max2play](#)

### Hilfreiche Dokumentationen:

[das ultimative Handbuch zum Thema Internetradio, immer aktuell](#)

[Adafruit Wifi Radio learning](#)