

## Raspberry Pi

2097618

# Digitalradio mit Raspberry Pi

01.07.2020 | 11:09 Uhr | Swen Hopfe

Gründlich überarbeitet gibt es das Digitalradio mit Raspberry Pi nun in der dritten Version.



[Vergrößern](#) Digitalradio mit Raspberry Pi

© Swen Hopfe

Wir starteten im Februar 2015. Die Idee war, auf Basis des [Raspberry Pi](#) ein eigenes Digitalradio zu bauen, das mit möglichst vielen Quellen umgehen kann. Am Anfang nutzten wir noch einen Pi der ersten Generation, einen “Hifiberry”-DAC und SDR-Stick. Danach zogen Pi2 und eine DAB-Pi-Platine ein.

Im aktuellen Projekt verbauen wir wieder neue Hardware. Auch die Bediensoftware wurde komplett erneuert, am grundlegenden Konzept aber haben wir festgehalten. So wollten wir wieder Material aus dem heimischen Netz und von Datenträgern abspielen sowie Webradio streamen und über Antenne empfangen können. Außerdem sollte eine frische [Optik](#) her und alles leichter bedienbar sein.

Das Gehäuse kommt nun von einem ausgedienten [Radio](#), wo wir sämtliche Elektronik und die Gehäuselautsprecher entfernt haben.



© Swen Hopfe

### [Vergrößern](#) Gehäuse

© Swen Hopfe

5 Gründe FÜR das iPhone 11, 11 Pro & 11 Pro Max

Die Stromversorgung soll jetzt über ein externes Netzteil erfolgen. So ist genug Platz für alle unsere Komponenten.

Im Wesentlichen sind das der [Raspberry Pi](#), wo wir einen 3B+ verbaut haben, Hifiberry DAC+ ADC, Monkeyboard für FM- und DAB+, ein Audioverstärker-Board auf Basis TPA3116, Spannungswandler, TFT-Touchpanel und ein kleines OLED für die Anzeige an der Front.



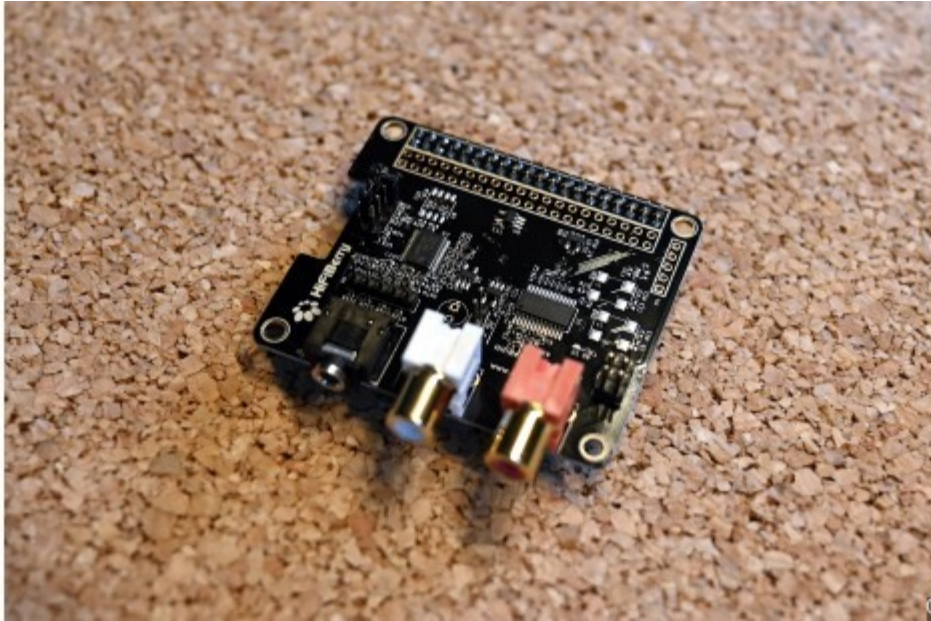
© Swen Hopfe

### [Vergrößern](#) Monkeyboard

© Swen Hopfe

Das Monkeyboard ist unsere aktuelle Wahl für den Empfang per FM und DAB+. Mit dem Pi ist das Board per USB verbunden. Die softwaremäßige Ansteuerung basiert auf Anleihen aus dem Github-Projekt vom Nutzer "schlizbaeda". Eine tolle Arbeit.

Wir haben dort die Keystone-Firmware für das neue Pro-Board ersetzt und auf die [digitale](#) Verbindung per I2S-Audio zum Hifiberry verzichtet. Die soll einzig nur der Raspi nutzen. Das Monkeyboard sendet bei uns also nur analog über die Klinkenbuchse.



[Vergrößern](#) Hifiberry DAC+ ADC

© Swen Hopfe

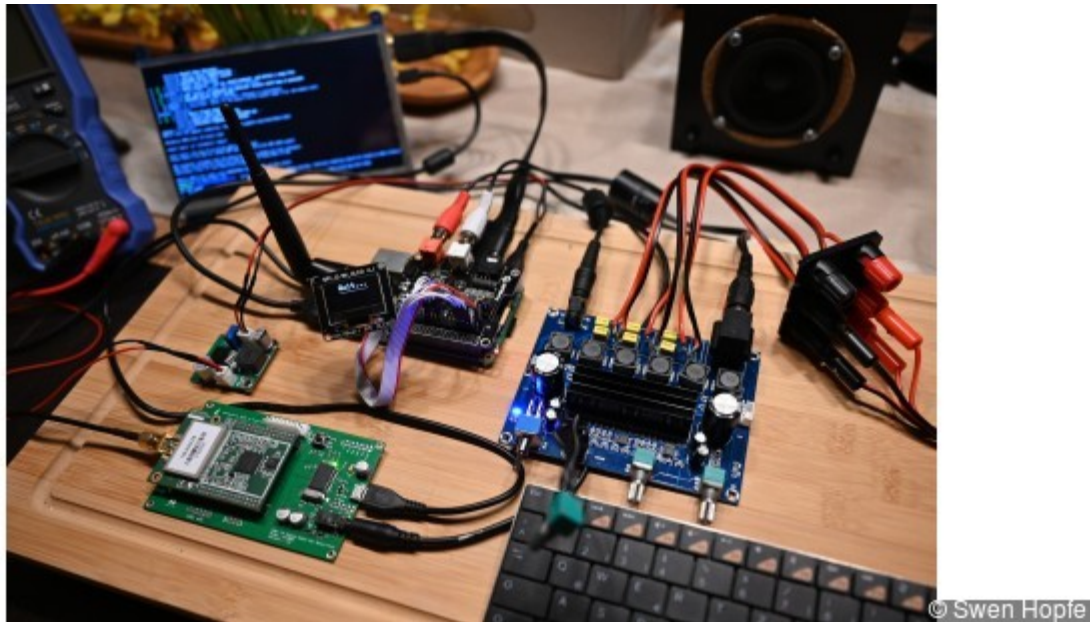
So gehen wir diversen Schwierigkeiten aus dem Weg, wie bei konkurrierenden I2S-Verbindungen aus mehreren Quellen umschalten zu müssen. Einen analogen wie digitalen Input bietet aus der Hifiberry-Serie aber nur das DAC+-ADC-Modell. Das ist also unsere neue Schaltzentrale und die gibt per Cinch-Buchsen an den Endverstärker weiter.

Damit dort nicht nur der Input vom Raspi ganz [digital](#) per I2S ankommt, sondern auch das Audiosignal auf dem Analogeingang, muss eine softwaremäßige Brücke geschaffen werden. Denn der Analoginput beim Hifiberry ist vordergründig für Record-Zwecke gedacht und deshalb auf dem Board nicht mit dem RCA-Ausgang elektrisch verknüpft. Da schafft "alsaloop" Abhilfe. Wann immer wir also FM oder DAB vom Monkeyboard wiedergeben wollen, ist ein

```
$ alsaloop -t 20000
```

in der Radio-Software vorangestellt, was den Loop von der Klinke zum Verstärkerausgang schafft. Das Tool ist bei der Installation der "alsa-utils" mit dabei.

Überhaupt sollten vor dem Einbau alle Module einzeln getestet, dann miteinander verknüpft und die Software schrittweise aufgebaut werden.



### Vergrößern Verkabelung für Tests

© Swen Hopfe

Zur Inbetriebnahme des Pi empfiehlt sich wie immer ein neues Image per Download und ein [Update](#) der Bezugsquellen und der Software. Aktuell ist bei uns ein “Buster” installiert, nebst den “alsa-utils”, der Software aus dem “dabd”-Projekt und der Bibliotheken für das OLED.

Für unser TFT-Panel am HDMI-Anschluss muss man noch die grafische Ausgabe anpassen. Das geschieht im Pi-OS in der “config.txt”, wo auch das Overlay für den Hifiberry eingetragen werden muss. Die wesentlichen Einträge für unseren Bildschirm sind:

```
hdmi_group=2  hdmi_mode=87  hdmi_cvt=1024 600 60 6 0 0 0  hdmi_drive=1
```

Und für die Konfiguration der Audiowiedergabe haben wir eingetragen:

```
dtparam=audio=off  dtoverlay=hifiberry-dacplusadc
```

Damit unsere Radio-Software von Beginn an startet, haben wir noch die “rc.local” auf dem Pi angepasst. Dort wird auch ein Netzwerkverzeichnis gemounted, auf dem man später bequem alle Audiodateien ablegen kann, welche zur Wiedergabe im LAN gefunden werden sollen.

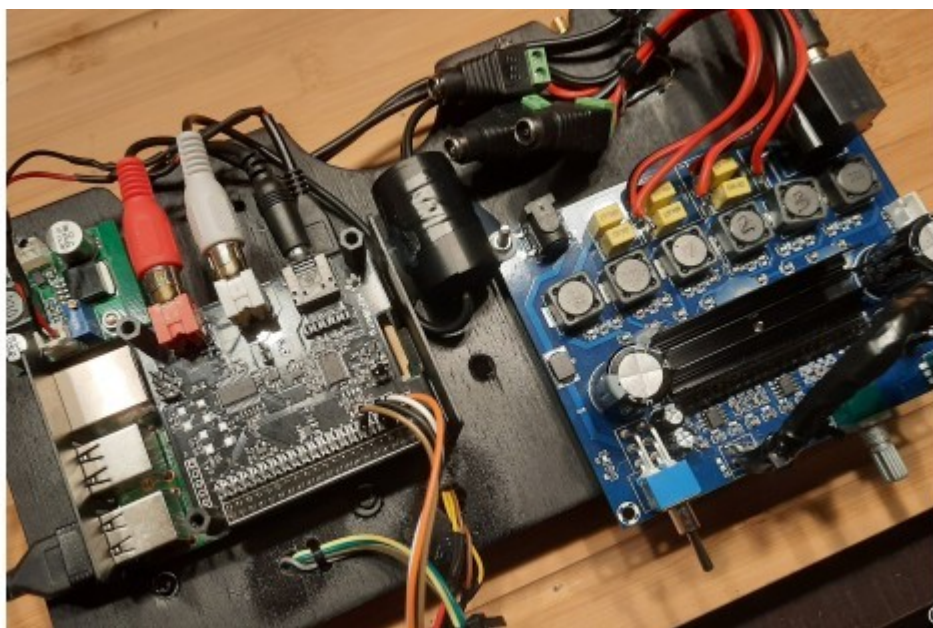
Am besten baut man alles Schritt für Schritt auf. Für die grafische Ausgabe dient uns “pygame” für Python, wo man keinen Linux-Desktop laufen lassen muss. Manches ist da etwas schwieriger zu realisieren als man von einem Fenstersystem gewohnt ist. Eine gute Wahl aber, wenn man nach einer flinken und robusten Anzeigemöglichkeit sucht. Zu Wartungszwecken kann man aus der Bedienoberfläche auch zum Linux-Prompt zurückkehren.



### [Vergrößern](#) Im lokalen Netzwerk

© Swen Hopfe

Während der laufenden Entwicklung verbindet man sich mit dem Pi wie gewohnt per *ssh* von einem entfernten Terminal aus. Um vom Arbeitsplatzrechner auch die Radiosoftware zu bedienen, loggt man sich mit X-Option ein, und man bekommt die vom Gerät gewohnte grafische Ausgabe. So kommt das Digiradio als App auf den Arbeitsplatz.



### [Vergrößern](#) Zwischenboden mit Modulen

© Swen Hopfe

Spielt alles zur Zufriedenheit zusammen, kann es an den Einbau gehen. Bei unserem Gehäuse geschieht der von vorn, wo die meisten Module auf einem Zwischenboden im Inneren verankert werden. Für das große Panel ist oben ein Ausschnitt vorgesehen, für Schalter und Regler hat die Frontblende Bohrungen bekommen.

Hinten sind die Lautsprecheranschlüsse nebst Buchse für die Stromversorgung angeordnet. Es reicht uns ein 12-V-Netzteil, was die Spannung für die Verstärkerendstufe mit Subwoofer-Ausgang mitbringt.

Die 5-V-Betriebsspannung für den Digitalteil wird durch einen DC/DC-Wandler erzeugt. Die versorgt auch den kleinen USB-Hub, auf dem zwecks besserer Verbindung die Wifi-Antenne außen angesteckt ist, genauso wie der Funkempfänger zur Fernbedienung.



#### [Vergrößern](#) Blick auf Rückseite

© Swen Hopfe

Nach dem Booten startet auf dem Pi unser Python-Skript. Bis das Gerät betriebsbereit ist, dauert es ein paar Sekunden, durch das OLED in der Front signalisiert. Jetzt steht auch das Bedienmenü auf der Gehäuseoberseite zur Verfügung.

Durch die einzelnen Ebenen kann man sich per Fingertipp, mittels Fernbedienung oder per Jog/Shuttle-Knopf bewegen. Dazu ist auf der Front ein Drehimpulsgeber verbaut und mit drei GPIO-Pins des Pi verbunden.



## Vergrößern Blick auf Vorderseite

© Swen Hopfe

Die drei Drehregler auf der rechten Seite bedienen die Endstufe, wo man die Lautstärke am Gerät einstellen kann, nebst Bass und Durchsatz zum Subwoofer, wenn man einen angeschlossen hat.

Unser Vorgänger hatte auch schon einen Bildschirm. Wo er doch die wichtigste Bedienerschnittstelle und das “Gesicht” vom Gerät ist, haben wir hier richtig viel neu gemacht. Zu Gunsten eines größeren Panels mit Touchbedienung sitzt der Bildschirm nun in der Gehäuseoberseite.

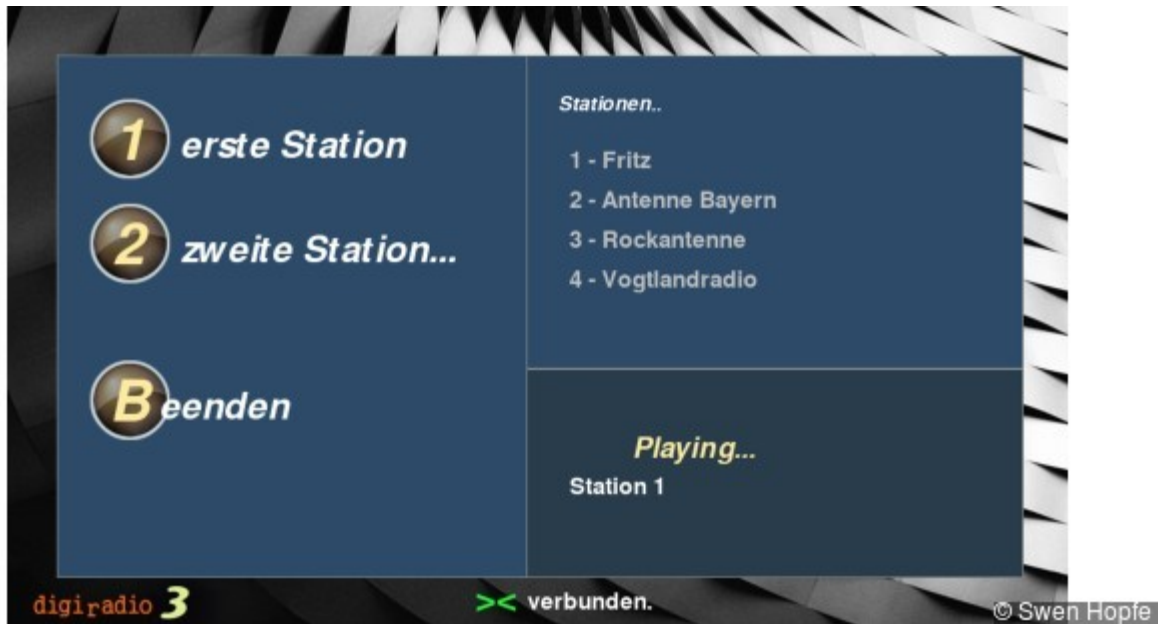


## Vergrößern Touchpanel auf der Oberseite

© Swen Hopfe

Aus dem Startmenü verzweigt man in die verschiedenen Wiedergabemodi, wie “Play aus dem Archiv” von lokalem [Netzwerk](#) oder Datenträger, “FM und DAB-Radio” per Antenne. Und “Internetradio”, also IP-Radio per Web.

Die gefundenen Titel können per Direktwahl oder Vor- und Rücklauf angesprungen werden. Und alles ist nun grafisch recht nett aufbereitet. Wird ein USB-Stick an der Außenbuchse angesteckt, kriegt man die Aktualisierung der Titel, in dem man das Menü ein zweites Mal aufruft.



### [Vergrößern](#) Menü IP-Radio

© Swen Hopfe

Daneben werden auch allgemeine Informationen in einer Statusleiste angezeigt. Beispielsweise, ob das Gerät überhaupt eine Internetverbindung aufbauen konnte. Schlägt das fehl, erhält man die betroffenen Bereiche nur zur Ansicht und ausgegraut. Manche Funktionen sind auf der Oberfläche undokumentiert gelassen, die soll den Bediener schließlich auch nicht überfordern und schön übersichtlich daherkommen.

Als Fernbedienung kann man jede Funktastatur verwenden, deren Sender dann an einem USB-Port des Pi steckt. Gegenüber einer Infrarot-Fernbedienung hat das den Vorteil, dass man den Pi auch erreichen kann, wenn die Radio-Anwendung nicht aktiv ist.



### [Vergrößern](#) Fernbedienung

© Swen Hopfe

Durch unser großes Display gelingt das direkt am Gerät ohne zusätzlichen Bildschirm. Möchte man nur die Radiosoftware bedienen, macht sich für den "Normalbetrieb" ein



kompaktes Modell gut, was dann wie eine TV-Fernbedienung daherkommt. Bei der ganzen Erneuerung sollten die Lautsprecher auch nicht nachstehen. Die alten Chassis tun es noch und passen ganz gut zum neuen Innenleben mit Mitteltönern von 15 Watt.



© Swen Hopfe

### [Vergrößern](#) Lautsprecher

© Swen Hopfe

In Kürze werden wir unser Projekt auch auf dem Github unter <https://github.com/swenae> veröffentlichen.

Multimedia-Projekte mit dem [Raspberry Pi](#) haben für mich einen besonderen Reiz, weil sie nicht nur “Proof of Concept”, sondern tägliche Begleiter sein können, die man zu Hause gerne nutzt.

So sah unser erstes Projekt aus:



© Swen Hopfe

### [Vergrößern](#) Das erste Digitalradio-Projekt

© Swen Hopfe

Ich finde, das neue “Digiradio” ist ein echter Hingucker mit toller Funktionalität geworden, schön kompakt und aufgeräumter als der Vorgänger. Klar, dass man da schon über die nächsten Erweiterungen nachdenkt. So sind ein passender Subwoofer geplant und eine reine Textversion für die Fernsteuerung per Terminal im [Web](#) zu Hause.



© Swen Hopfe

### [Vergrößern](#) Das neue “Digiradio”

© Swen Hopfe

Später soll es auch die Umstellung auf ein Read-only-SD-Karten-Image geben, bei der das lokale Songarchiv auf einen internen Plattenspeicher ausgelagert ist, so dass man auch ohne den Pi herunterzufahren das [Radio](#) einfach abschalten kann...

### **Hier finden Sie noch mehr Raspberry-Projekte von Swen Hopfe**

[Raspberry Pi Zero W: Kamera per WiFi fernsteuern](#)

[Fotoslider mit Raspberry Pi](#)

[So machen Sie den Raspberry Pi sicherer](#)

[Raspberry Pi: Kontrolle per Smartphone-App](#)

[Pidocor: Raspberry Pi überprüfen](#)

[Raspberry Pi: Computerschach per Pi-Cam spielen](#)

[NFC-Reader mit Raspberry Pi selbst bauen](#)

[Raspberry Pi Zero W als smarterer USB-Stick](#)

[Zeitraffer-Aufnahmen mit dem Raspberry Pi Zero machen](#)

[Webcam mit Raspberry Zero Pi W einrichten](#)

[Nächtliche Tierfotografie mit NoIR-Cam und Raspberry Pi](#)