

Die Installationsanweisung der Variante 1 soll in Kurzform beschreiben wie man einen USB RTL-Stick als lokalen Empfänger oder abgesetztem RX betreiben kann

Welche Software braucht man, um einen RTL-Stick zu nutzen?

Unter <https://www.rtl-sdr.com/tag/sdrsharp/> lässt sich die Software SDR# herunterladen
Im Unterverzeichnis der url <https://www.rtl-sdr.com/sdrsharp-plugins/> gibt es zusätzliche Plugins wie dsd+, CTCSS, DCS und TETRA

Auf dem heimischen PC oder Laptop erstellt man der Einfachheit halber z.B. ein Verzeichnis c:\sdrsharp und kopiert die aktuelle SW von SDR# dort hinein. (Beispiel Kopie auch im Download verfügbar) Hier sollte man sich noch Unterverzeichnisse wie DSD+ anlegen und die Treiber und die Bibliotheken von Mbelib dort hinkopieren. Innerhalb der SDR# sharp Konfigurationsfile, legt man die „magicline“ `<add key="DSD" value="SDRSharp.DSD.DSDPlugin,SDRSharp.DSD" />` mittels Editor zusätzlich an. Beim nächsten Aufruf von SDR# findet man einen zusätzlichen Menüpunkt für DSD. Die SDRsharpDSD.dll muss man noch händisch in das Verzeichnis kopieren. Will man auch TETRA Signale, die nur unverschlüsselt decodiert werden können, benötigt man ebenfalls eine magicline. z.B.

```
<add key="DSD" value="SDRSharp.DSD.DSDPlugin,SDRSharp.DSD" />
<add key="Tetra" value="SDRSharp.Tetra.TetraPlugin,SDRSharp.Tetra"/>
```

Der nächste Schritt ist das Herunterladen und installieren von vb-cabel driver unter <https://www.vb-audio.com/Cable/> . Diese SW macht eine Umleitung der NF, die default mäßig z.B. über den PC Lautsprecher ausgegeben wird. Das ist erforderlich, um den DSD Decoder mit NF zu versorgen. Innerhalb von DSD stellt man sich dann die Ein- und Ausgabe so ein, daß der DSD über VB-cable versorgt wird und im Ausgang die Lautsprecher nutzt. Steckt man zum test den RTL-Stick direkt über USB in den PC, muss man innerhalb des SDRsharp Verzeichnisse das Tool install-rtl.bat benutzen, um den RTL-Stick von den Windows Treibern zu befreien und die Treiber für SDR# benutzen zu können.

MBE-LIB und RTL-Stick direkt am PC benutzen ohne Weiterleitung:

Download und Kurzanleitung in Schritten:

1. <https://airspy.com/download/> hier Windows SDR Community Package laden, sonst zu 2.
2. <https://www.rtl-sdr.com/tag/sdrsharp/> download sdr# falls auf airspy nicht verfügbar
3. <https://www.vb-audio.com/Cable/> download vb-kabel driver cable
4. <https://www.dsdplus.com/download-2/> download dsdplus und dsdplus dll files
5. Dsd plus dll files in das Verzeichnis dsdplus kopieren
6. Sdr# auspacken und die .dll files von dsd (SDRsharp und MagicLine kopieren)
7. Magicline text in Plugins.xml einfügen
8. `<add key="DSD" value="SDRSharp.DSD.DSDPlugin,SDRSharp.DSD" />`
9. `<add key="Tetra" value="SDRSharp.Tetra.TetraPlugin,SDRSharp.Tetra"/>` kopieren
10. Vbcable installieren
11. Sdrsharp starten rtl stick anstecken und software von Windows installieren lassen
12. Im Verzeichnis sdr-sharp das programm install-rtl.bat ausführen (ersetzt die Windows Treiber durch passende Treiber für sdrsharp)
13. In SDRsharp Audio zu Windows Direct, sound virtual Cable Input setzen
14. Unter SDRsharp DSD Interface Audio ebenfalls virtual Cable input wählen, dann configure Vbcable unter Geräte Sound auf Virtuell Cable ändern Input Sound

Die Installationsanweisung der Variante 1 soll in Kurzform beschreiben wie man einen USB RTL-Stick als lokalen Empfänger oder abgesetztem RX betreiben kann

Von der Raspberrypi Homepage sollte man das aktuelle Fullimage für den Raspi herunterladen.

<https://www.raspberrypi.org/downloads/raspbian/>

Mit z.B. der Software Balena etcher das heruntergeladene image installieren auf einer mindestens 16 GB großen SD Karte. Auf der neu erstellten sd-karte, im Verzeichnis boot, eine leere Datei mit dem namen ssh erstellen, dass macht alles einfacher, denn dann funktioniert sofort der ssh Zugang mittels Putty.

Installieren der RTL Pakete auf einem RasPI:

Einloggen als pi;

passwd: Image default (raspberrypi)

sudo su und passwort ggf. ändern oder über raspi-config password, vnc etc. setzen.

```
apt install rtl-sdr librtlsdr-dev
```

```
apt-get install git cmake libusb-1.0-0.dev build-essential
```

```
git clone git://git.osmocom.org/rtl-sdr.git
```

```
cd rtl-sdr/
```

```
mkdir build
```

```
cd build/
```

```
cmake ../
```

```
make
```

```
sudo make install
```

```
sudo ldconfig
```

```
cd ..
```

```
cp rtl-sdr.rules /etc/udev/rules.d
```

Ein erster Test kann händisch so erfolgen: `rtl_tcp -a 192.168.176.46 &`

(Anmerkung die IP-Adresse ist die von eth0: abfragen z.B. mit ifconfig)

oder für den automatischen Betrieb im folgendem Verzeichnis:

`etc/systemd/system/rtl_stick.service`

Den `rtl_stick.service` file mit folgendem Inhalt generieren :

```
#####  
[Unit]  
Description= RTL_Stick Service  
After=multi-user.target network-online.target  
Wants=network-online.target  
  
[Service]  
Type=idle  
ExecStart=/bin/bash -c "/usr/bin/rtl_tcp -a $(/sbin/ip -o -4 addr show dev eth0 | awk '{ print $4; }' | cut -d/ -f1) "  
  
[Install]  
WantedBy=multi-user.target  
#####
```

Den Service danach installieren mit:

```
systemctl enable rtl_stick.service
```

```
systemctl daemon-reload
```

Die Installationsanweisung der Variante 1 soll in Kurzform beschreiben wie man einen USB RTL-Stick als lokalen Empfänger oder abgesetztem RX betreiben kann

URL für den Download der RaspberryPI Software Variante 1

http://26731.webhosting8.1blu.de/FA/Image_Variante1/webrx_V1_test123.img.gz

Image variante 1: (3.7 GB)

webrx_V1_test123.img.gz
usr: root und pi
passwd: test123

URL für den Download der sdr# PC Software mit installierten dsd+ und TETRA Plugin

http://26731.webhosting8.1blu.de/FA/Image_SDR/sdr-sharp.zip

URL für die Variante 2

<https://www.openwebrx.de/download/rpi.php>