BoomBox

ThirtySomething

March 30, 2019

Contents

1 Motivation

There are many ways for parties to play music. One of the modern variants of this is streaming via a smartphone. However, this requires a functioning WLAN or mobile phone reception. If you want to use the telephone network, you need a corresponding data connection or the corresponding data volume. But what do you do if neither one nor the other is given? Then you use the *BoomBox*. This is a kind of modern ghetto blaster. The device is independent of smartphone and WLan or mobile phone reception.

2 The device

The device *BoomBox* consists of two components, the hardware component and the software component.

2.1 The hardware component

On the one hand, the system should be cost-effective, but on the other hand it should not be old-fashioned. The following components meet these requirements:

- A Raspberry Pi 3 B+.
- An Pi-DigiAMP+.
- A Raspberry Pi Touch Display.
- A M.2 to USB Adapter.
- A 250 GB M.2 SSD.
- A Power supply.
- Two USB cable.
- A MicroSD card.
- A Step Down Converter.
- An Adatper with terminal block.
- A Cable with plug.

The Raspberry Pi is an inexpensive single board computer that is perfect for this project. The sound card greatly improves audio output – the Raspberry Pi isn't so convincing here by nature. The original touch display is used for operation without additional input devices. On Parties can be a bit wild every now and then. So that shocks don't have any influence on the *BoomBox*, instead of a conventional hard disk an SSD is used as mass storage. This is connected to the system via an adapter.

2.2 The software component

The software essentially consists of only one point: the music distibution Volumio. This distribution comes with support for the above hardware. Of course some fine-tuning is necessary to simplify operation and handling. This is described in the chapter 5.3.

3 Preparations

Before the entire system can go to the start, a few preparations have to be made.

3.1 Updating the Raspberry Pi firmware

We'll connect the Raspberry Pi with a network cable. Then we download the latest version of Raspbian. This is the normal operating system for the Raspberry Pi. We write the image with the Win32 Disk Imager on a MicroSD card. To allow access via SSH, we create the empty file *ssh* in the boot partition. We boot the Raspberry Pi and connect to *ssh*. Then we enter the following commands:

```
sudo apt-get update
sudo apt-get install git
sudo wget https://raw.github.com/Hexxeh/rpi-update/master/rpi-update -O
    /usr/bin/rpi-update && sudo chmod +x /usr/bin/rpi-update
sudo rpi-update
sudo reboot
```

Figure 1: Firmware Update

3.2 SSD mounting

The SSD must be mounted on the adapter. This is very simple - insert the SSD into the slot, tighten the screw and you're done. The result looks like

this:

Figure 2: SSD with adapter

3.3 Prepare SSD

We format the SSD with ext4. This means that the SSD can no longer be used directly on the Windows PC. However, the file system is more robust than FAT32. This is especially true in the event of sudden power loss. If you later include the *BoomBox* in your own network, you get access to the SSD.

3.4 Filling up the SSD

To be able to offer Volumio also a music selection at the start, the SSD is initially refuelled. For this a SMB-Share is connected and the files are copied. This step must only be performed once.

```
# Utilities to mount the SMB-Shares
sudo apt-get install cifs-utils
# Mount SMB-Share
mount -t cifs -o user=<smbuser>,domain=<domain|workgroup> //<IP of the share>/<sharename>
    /mnt
# Create mountpoint /music for the SSD
sudo mkdir /music
# Mount the SSD to /music
sudo mount /dev/sda1 /music
# Start filling up of the SSD
sudo cd /mnt
sudo cp -R * /music
```

Figure 3: Filling up the SSD

After the copy process is finished, the Raspberry Pi can be shut down. We remove the MicroSD card. For the next use we save Volumio on it.

4 The hardware component

This is about the mechanical assembly of the *BoomBox*. Since everything is stacked on top of each other, I'm also talking about the *Hardwarestack*.

4.1 The display and the Raspberry Pi

We'll start with the display. When unpacking, it is noticeable that the control board is already connected and mounted.

Figure 4: Display

This simplifies assembly for us. How this is done is simply explained in this YouTube Video.

Caution: For the *BoomBox* we only connect the ribbon cable. In the video, the Raspberry Pi is fastened with screws. Instead of these screws we use spacer bolts M2,5 x 11mm. After the Raspberry Pi the sound card and the converter board for the SSD will be added.

Figure 5: Display with Pi

4.2 The sound card

There's not much to explain here. The sound card is placed on the GPIO bar of the Raspberry Pi. Then the spacer bolts, which were supplied with the SSD adapter board (!), are screwed on for fixing.

Figure 6: Display, Pi and iQAudio

4.3 The converter board

At the end comes the SSD with the converter board. We have already connected both in the preparations. This board is fixed with screws on the spacer bolts of the sound card. The Raspberry Pi is powered by the soundcard. However, this is too less to supply the converter board with the SSD via USB. Therefore we have to set the jumper PWR_U so that the middle pin and the pin closest to the board edge are bridged. This ensures that the converter board is not supplied with power via USB, but via the extra input.

Figure 7: Display, Pi, iQAudio and SSD

4.4 Adapter cable

The power supply has only one output with 19V. For the display and the converter board 5V are required. For this we need an adapter cable. The cable has a socket, into which the plug of the power supply comes. This socket has screw terminals on the other side. We connect two cables to these screw terminals. One, which again has a plug for the sound card. And one which is connected with screw terminals to a so-called Step Down Converter. This Step Down Converter has two USB ports, which we use to power the display and the converter board.

Figure 8: Adapter cable

4.5 The result

If everything was assembled correctly, it looks like in the following picture.

Figure 9: Hardwarestack

And if the cables were also connected, it looks that way:

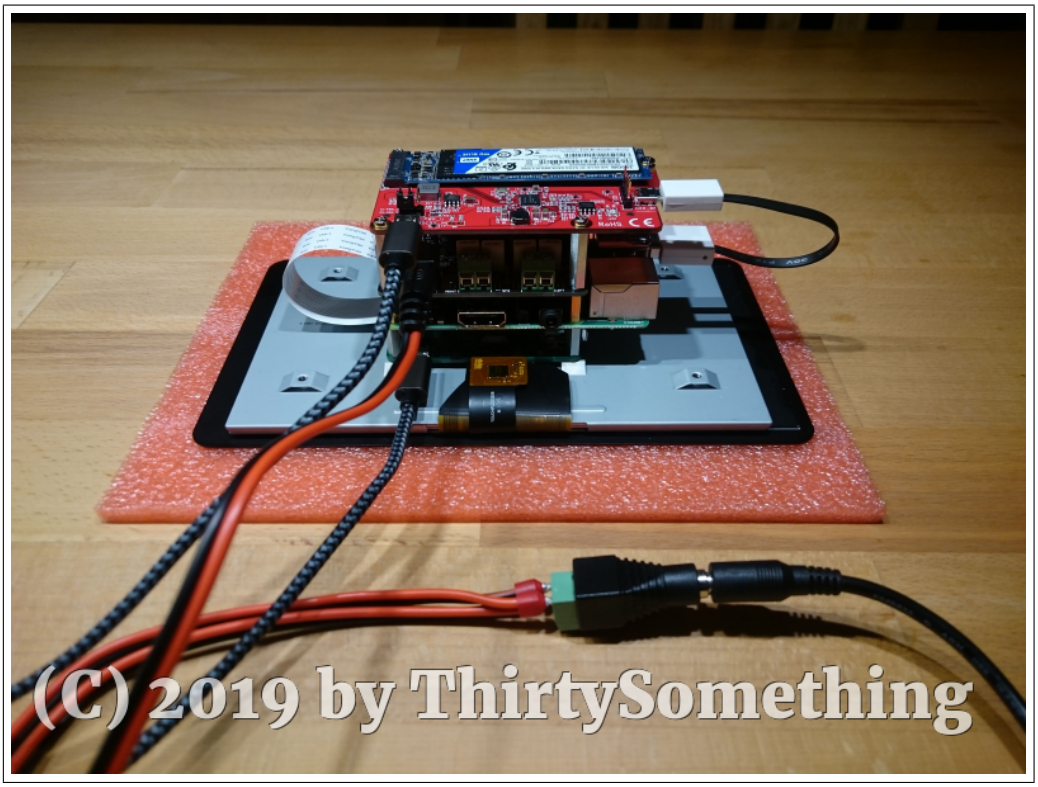

Figure 10: Hardwarestack with cables

5 The software component

This is about the installation and configuration of Volumio.

5.1 First installation

Prerequisites for the installation is the Filling up the SSD. And of course the assembly of the hardware stack. For this we download the image of Volumio. After that we try the Win32 Disk Imager again and save the image on the MicroSD card. After the image is installed, the card is inserted in the Raspberry Pi, we start the system. Please make sure that the Raspberry Pi is connected to the network with a network cable.

5.2 The plugins

We can find out the IP address of the *BoomBox* via our router. Then we call up the IP address in the browser. The start screen will look like this.

Figure 11: Initial screen

By the time we see this image, we have already made a great deal of progress. In order for the touch screen to work, an appropriate plugin is required. To do this we go through the settings – the gear in the upper left corner.

Figure 12: Settings

For the plugins we select the corresponding menu item. The plugin for the touchscreen can be found under *Miscellanea*, it is called *Touch Display Plugin*.

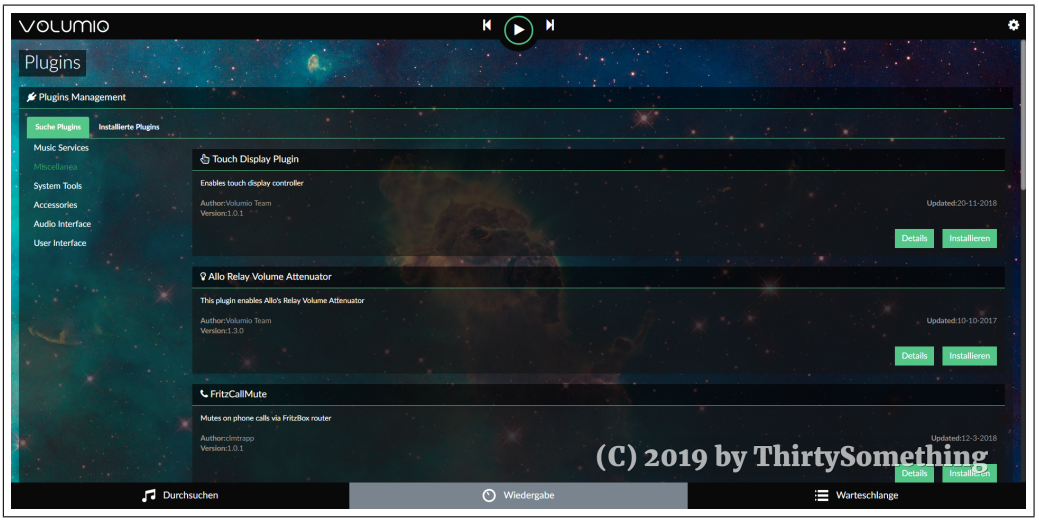

Figure 13: Touchscreen

Another plugin is a simple equalizer. We install it as well. It can be found under *Audio Interface*.

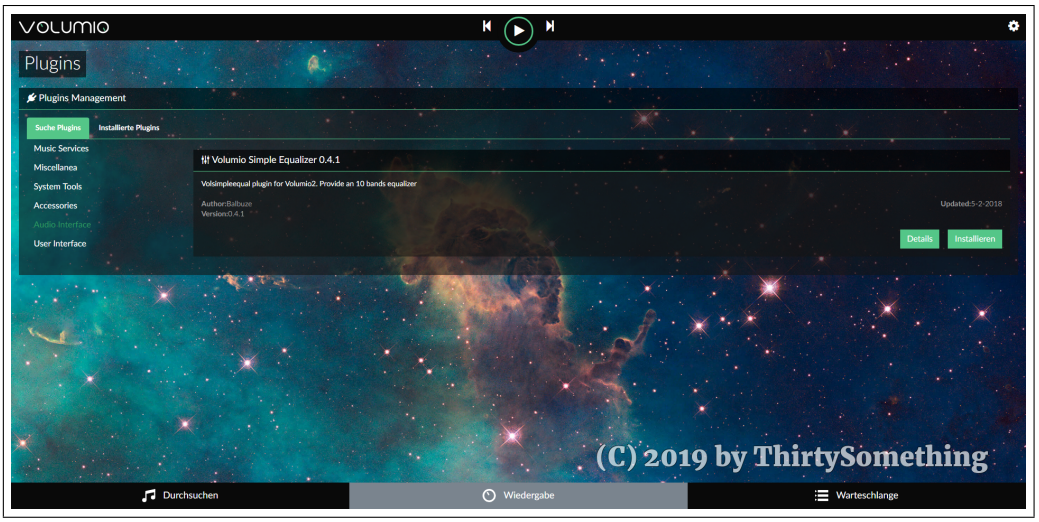

Figure 14: Equalizer

After the plugins have been installed, you have to activate them. This is done on the second tab *Installed Plugins*. After they are activated, it looks like this.

Figure 15: Plugins

5.3 Fine-tuning

Now it's time for some fine tuning. How to do them is explained on this page.

Note: After performing one or more of these configuration steps, a reboot is necessary. Only then the changes will take effect.

5.3.1 The mouse pointer

We start by hiding the mouse pointer. First we have to activate SSH. This can be done via the browser with the following URL:
http://<IP-the-BoomBox>/dev – in my case for example with *http://192.168.2.17/dev*. On this page we find buttons to activate and deactivate access with SSH. For our purpose we need this actively.

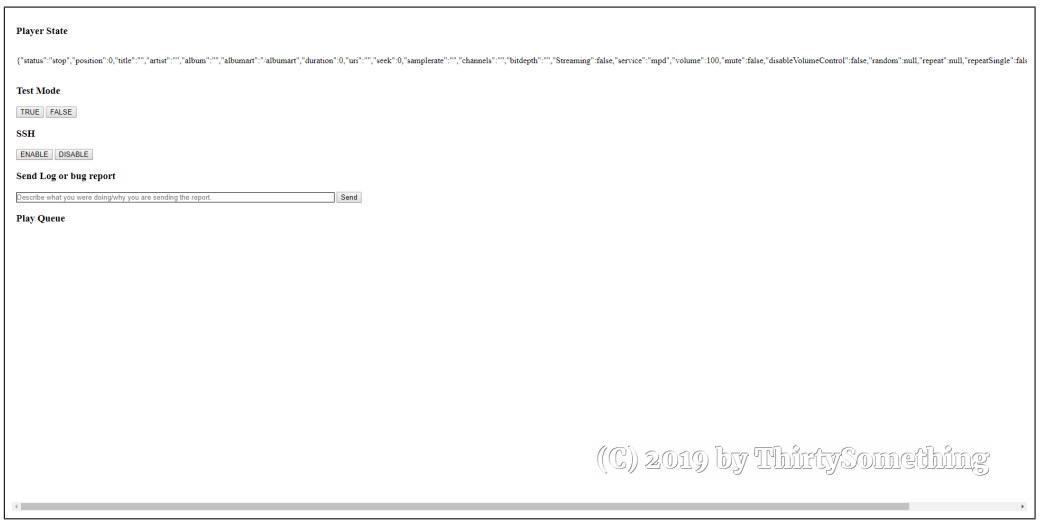

Figure 16: SSH

Then we log on to the system via ssh. The username is *volumio*; the password is identical to the username. Then we edit the file configuration file for the kiosk mode. We add `-- -nocursor` to the line.

```
sudo nano /lib/systemd/system/volumio-kiosk.service
# Original line
# ExecStart=/usr/bin/startx /etc/X11/Xsession /opt/volumiokiosk.sh
# Modified line
ExecStart=/usr/bin/startx /etc/X11/Xsession /opt/volumiokiosk.sh -- -nocursor
# Leave the editor by pressing CTRL+X
```

Figure 17: Kiosk mode

5.3.2 Screensaver

From time to time the display is simply *switched off*. That is, it becomes completely black. To prevent this, the following steps are necessary:

```
sudo nano /opt/volumiokiosk.sh

# Original lines
# xset +dpms
# xset s blank
# xset 0 0 120

# Adjusted lines
xset -dpms
xset s off
#xset 0 0 120
# Leave the editor by pressing CTRL+X
```

Figure 18: Screensaver

5.3.3 Access from Windows

At Volumio a samba is already installed by default. This allows easy access to the storage. However, the device offers different storage locations. This could cause confusion. That's why we make sure that only storage that is connected via USB can be accessed. So we can access the SSD from Windows without guesswork.

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

# Original lines
[Internal Storage]
       comment = Boombox Internal Music Folder
       path = /data/INTERNAL
       read only = no
       guest ok = yes

[USB]
       comment = Boombox USB Music Folder
       path = /mnt/USB
       read only = no
       guest ok = yes

[NAS]
       comment = Boombox NAS Music Folder
       path = /mnt/NAS
       read only = no
       guest ok = yes

# Adjusted lines
#[Internal Storage]
#       comment = Boombox Internal Music Folder
#       path = /data/INTERNAL
#       read only = no
#       guest ok = yes

[SSD]
       comment = Boombox SSD Music Folder
       path = /mnt/USB
       read only = no
       guest ok = yes

#[NAS]
#       comment = Boombox NAS Music Folder
#       path = /mnt/NAS
#       read only = no
#       guest ok = yes

# Leave the editor by pressing CTRL+X
```

Figure 19: Share

Under Windows, the device can then be accessed under the name `\\boombox` in Windows Explorer.

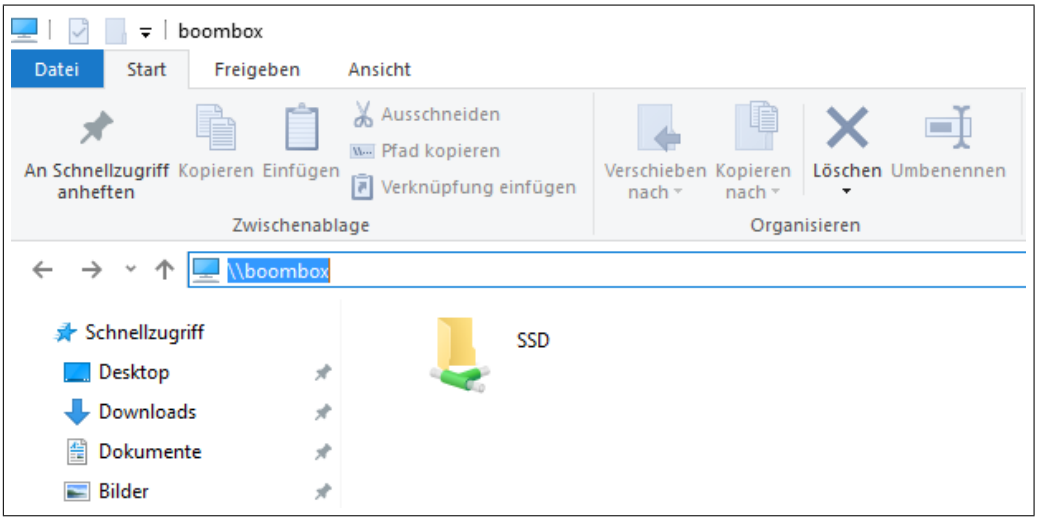

Figure 20: Access from Windows

Finally, we turn on the *SSH* access again. To do this we call up the corresponding page. See also chapter The mouse pointer.

List of Figures

Index