

DQMusicBox

A DIY MAKER project, making music accessible to dementia sufferers

20 September 2015



DQMusicBox allows some people with dementia to listen to and control their favorite music. It uses a familiar car radio user interface to control what is effectively a big MP3 player.

That's my Dad, DQ, in the photo. I was inspired to make a music box for him when I read a review of the documentary [Alive Inside](#). The documentary suggests that some people with dementia come alive when listening to their favorite music. Following that suggestion, I put DQ's favorite music on an iPad with good headphones. There was sustained joy on his face – joy from the familiar music and joy from the fact that nothing else in the frustrating world mattered for a few minutes. My Dad is doing pretty well at home. But he can't operate their big stereo or their iPad. But he can use DQMusicBox without assistance.

DQMusicBox is a fun maker project. Laser cutting, optional 3D printing, soldering, Python code, and a Raspberry Pi. But most importantly, there is music – a personalized selection that you choose on behalf of whomever you are gifting this to.

Now let's talk about you. I'm assuming that you are a nerd. Like me. This was my first time laser cutting, 3D printing, and working with a Pi. So if you are a nerd, you can do this. Once you have the parts you can do this in about a day, depending on your soldering skills and Pi skills.

I'm assuming that you will be giving this to someone who cannot use other music players. The goal is self-sufficiency. And the principal tactic is to use a familiar and simple user interface, modeled on old car radios.

I could not have done this without help. People were very generous with their time, and I really enjoyed the experience. This is almost certainly an incomplete list: Alex & Mike & others at [Ada's](#), the super smart staff at [Metrix](#), neighbor Randy, [Stephen Christopher Phillips](#), [Bob Rathbone](#), [Stephen Rusk](#), support at [Ponoko](#), and my son.



1 No warranty

USE THESE DQMUSICBOX PLANS AND SYSTEM AT YOUR OWN RISK. THE DQMUSICBOX PLANS ARE PROVIDED AS IS WITHOUT WARRANTY OF ANY KIND EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PLANS AND SYSTEM IS WITH YOU. SHOULD THE PLANS OR SYSTEM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. USE OF ANY TOOLS IS AT YOUR OWN RISK. IN NO EVENT WILL ANY PARTY BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PLANS OR SYSTEM.

2 Use cases

2.1 End user use cases

There are deliberately only three end user use-cases:

Name	Description & implementation
Start song	Turning either of the knobs will start music playing.
Change song	Turn the songs knob.
Change volume	Turn the volume knob.

There is no screen. This is by intent. The songs knob is the only way to select music. This is also by intent. It is based on an underlying assumption is that the end user does not need much music. One of the few benefits of dementia and memory issues is that you may never get tired of your favorite music. My Dad is quite happy with ten albums. He has not asked for more music. Thus the songs knobs is a sufficient for scrolling.

2.2 Caretaker use-cases (that's you)

You have the power of an Internet connected Raspberry Pi at your disposal. Here are things that I tend to do:

Name	Description & implementation
Pause	Tap the volume knob. Note that this also happens automatically – music pauses if there are no knob events in one hour.
Shut down	Long hold (15-30 seconds) on the volume knob.
Reboot	Long hold (15-30 seconds) on the songs knob.
Troubleshoot	I just use PuTTY . Note that the DQMusicBox system verbosely logs to /var/log/dqmusicbox.
Add/remove music	As per above, you probably won't do this often. I use SFTP via WinSCP .
Maintain Raspbian	I just use PuTTY. You could also use RDP or VNC.

My Dad has been using a DQMusicBox for six months, and so far no troubleshooting. But I did add some jokes that my son recorded, and the remote access worked as expected for doing this. I use the logmein service to access my Dad's old computer and Putty from there to his DQMusicBox.

3 The parts

Here are the parts.

- The wood case – see below (~\$50)
- From [Adafruit](#) (~\$60)
 - 1 [Raspberry Pi 2](#)
 - 1 [plastic case base](#)
 - 1 pack of [indicator LEDs](#)
 - 1 [USB audio adapter](#) (substitution not recommended as config is specific to this product)
- From [Vetco](#) (~\$14):
 - 3 [rotary encoders VUPN7453](#) (need two but mistakes can happen in soldering, mounts well to 3.5mm wood)
 - 2 [10K Ohm resistor NTE-QW310](#) (you only need one, but just in case)
 - 1 [panel mount headphone jack PH 45-233B](#)
 - 1 [heat shrink tube 4' x 3/16" CARRY-3/16HS](#)
 - I recommend calling Vetco (+1 425.641.7275) to place to place the order, and ask for the sealant heat shrink with a 3:1 shrink ratio. With this stuff, I didn't have to remove the male end of the jumper wire – just slid heat shrink over the male end to heat shrink over the jumper-coder connection.
- From [Amazon](#) (~\$55 w/o headphones, ~\$95 - \$135 with headphones)
 - 1 [16GB micro-SD card](#)
 - 1 [wi-fi](#) adapter (not needed if the device can live on built-in Ethernet)
 - 1 [case mount LED holder](#)
 - 1 [power supply \(USB\)](#) (you may already have one of these)
 - 1 [short headphone cable](#)
 - 1 [roll of sticky back velcro](#)
 - 1 [set of M-F jumper wires](#)
 - 1 set of good headphones e.g. over-the-ear [Sony MDR7506](#) (my preference, used for testing, ~\$80) or on-the-ear [Shure SRH144](#) (~\$40)
 - 1 spray can of [matte clear protective finish](#)
- Tools that you will need
 - Soldering iron & solder
 - USB keyboard (temporary)
 - Monitor or TV with HDMI input (temporary)
 - Wood glue e.g. [Elmer's](#)
 - Something to hold the case pieces in place while the glue dries – clamps or [painter's tape](#) or bookends
 - A small wood file, in case the wood knobs do not quite fit over the rotary encoder shafts

4 Make the case

You have some choices in how to turn the included plans into a laser cut wood case:

Option 1 – easy, mail order maker space, ~\$50

Takes ~10 minutes and the case pieces will arrive in ~10 days. Ponoko (mail order maker space) charges ~\$50 + tax + shipping. I did the later revs on the case via Ponoko, in part to give you a convenient option. In full disclosure I'll note that a friend is an investor in Ponoko, not that I'm expecting this project to generate a lot of business for them. Ponoko ships to the United States and New Zealand.

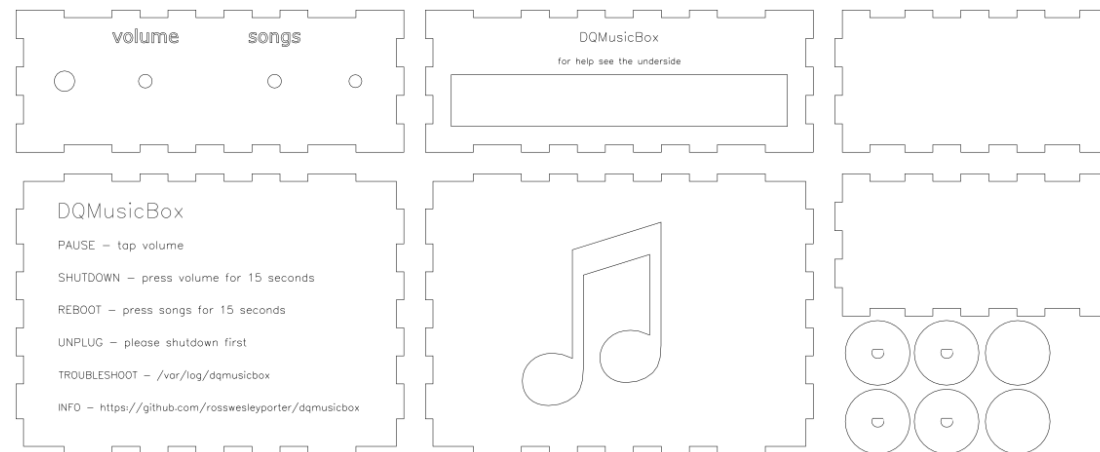
To order: <http://www.ponoko.com/design-your-own/products/dqmusicbox-case-12685>, and choose “Add to Personal Factory”. If prompted, choose Premium Veneer MDF – Maple, size is 24'x12'.

Option 2 – fun, your local maker space, ~\$50

Many cities have a maker space. I went to [Metrix](#) in Seattle, which was a lot of fun for a nerd like me. I got lots of great help. If desired, you can customize the plans e.g. add the recipient's initials. The plans are in the same github repository as this document. I used [Inkscape](#) (free) to create the plans. Adobe Illustrator also works well. The plans assumes a 3.5mm thick wood product. If you want to use a different wood thickness, then you may want to create a new set of plans via <http://www.makercase.com/>, and then paste in elements from my plan.

To find a local maker space, hit Google or see this directory: <http://spaces.makerspace.com/makerspace-directory>

In any case, pun intended, the result is a pile of precisely cut wood pieces that you glue together. Here is a preview of what those pieces will look like:|

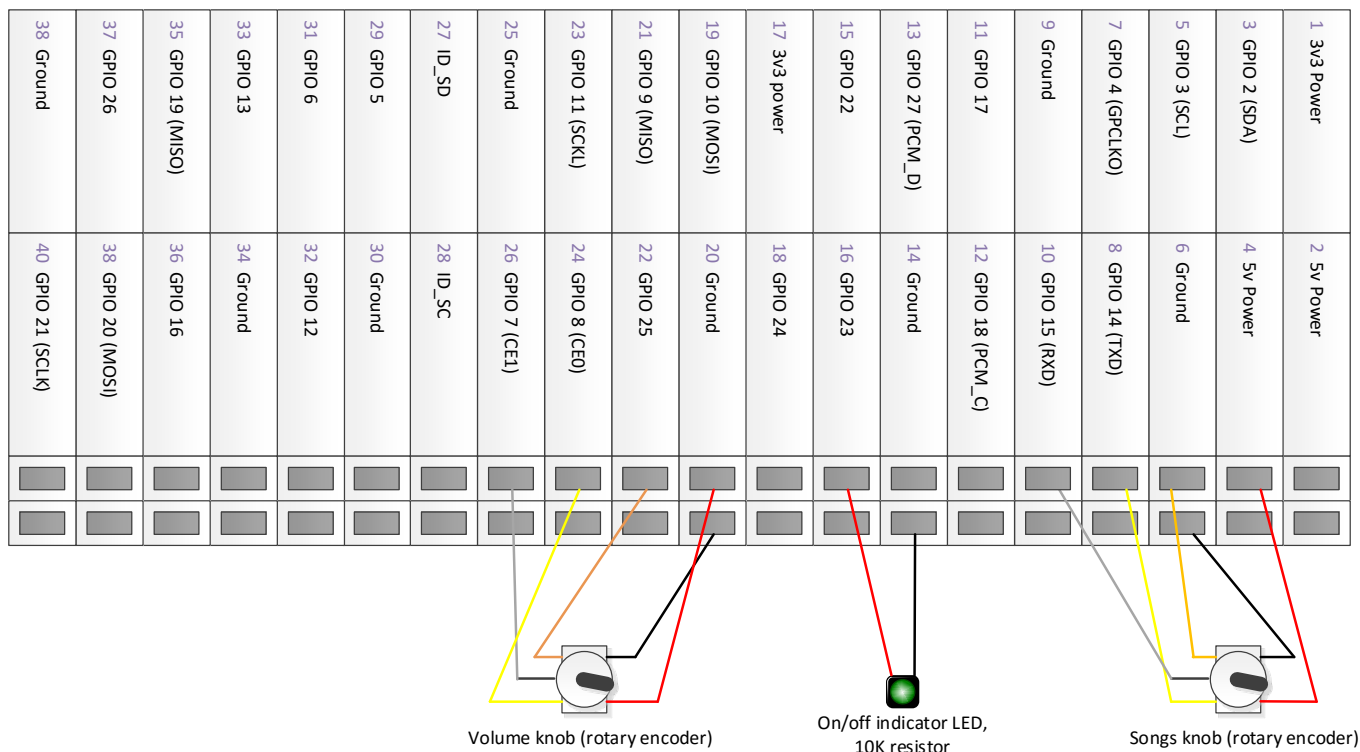


5 Prepare the wiring

Now it is time to solder.

1. **Rotary encoders.** Solder M-F jumper wires to the rotary encoders as shown below. Solder the M end to the rotary encoder, leaving the F end for connecting to the Pi later. There isn't much space between the rotary encoder pins, so soldering skill is needed. I bought some staff labor from my local maker space to do this. I watched and was able to solder later rotary encoders myself.
2. **Resistor for LED.** I put the resistor in-line with the ground/black wire. Note that you won't be able to solder the LED to the jumper wires until you mount the LED to the front panel in a later step.

Later, you'll use the wiring diagram below to connect the rotary encoders and the LED to the Pi.



6 Setup the Pi

6.1 The basics

There are many guides for the following, so they are not repeated here:

- Install Raspbian (see this [guide](#))
- Configure wi-fi (see this [guide](#))

6.2 Set the USB audio device as the default

Linux sound is wonderfully complicated. Below is the simplest method that I have found for setting the default audio device to be USB audio.

First, disable the built-in sound device so the USB audio device is the only remaining audio device:

```
sudo nano /etc/modules
Change:      snd-bcm2835
To:          #snd-bcm2835
```

Now remove the default override of USB audio:

```
sudo nano /etc/modprobe.d/alsa-base.conf
Change:      options snd-usb-audio index=-2
To:          #options snd-usb-audio index=-2
```

Now load XWindows, then immediately exit XWindows:

```
startx
[Menu.. Shutdown.. Logout]
```

While the above simplifies the audio configuration, it does cause a problem for a panel item in the startx desktop of recent versions of Raspbian. To remove that unhappy audio control panel item, comments out the five lines associated with the volumealsa plugin:

```
sudo nano ~/.config/lxpanel/LXDE-pi/panels/panel
#Plugin {
#    type=volumealsa
#    Config {
#    }
#}
```

Reboot

```
sudo reboot
```

Verify with `aplay -l` that the USB audio device is the only device shown.

```
aplay -l
```

For more information above audio changes: [disable the built-in sound device](#), [remove default override of USB audio](#), [panel](#)

6.3 Install/clone dqmusicbox

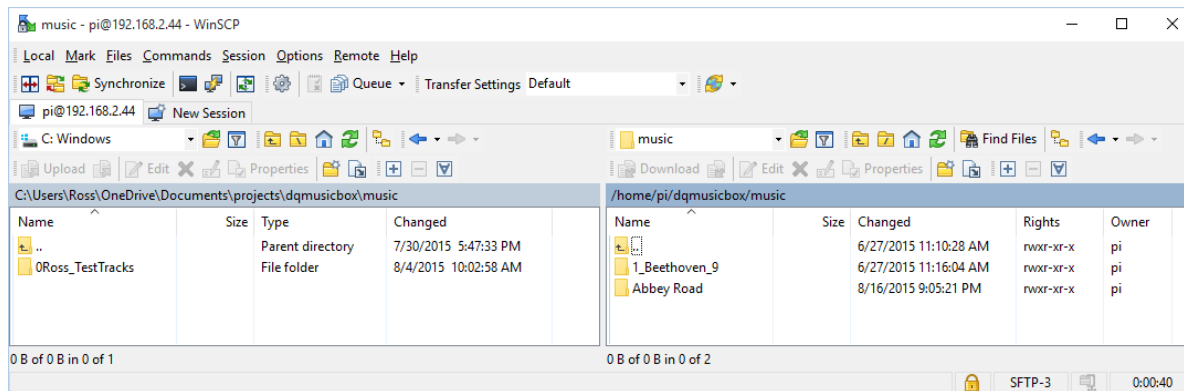
```
cd ~
git clone https://github.com/rosswesleyporter/dqmusicbox/
sudo chmod 755 ~/dqmusicbox/bin/dqmusicbox.py
```

6.4 Install VLC (music player) and Python bindings for VLC

```
sudo apt-get install vlc
cd ~
git clone git://git.videolan.org/vlc/bindings/python.git
cp ~/python/generated/vlc.py ~/dqmusicbox/bin
```

6.5 Add personalized music

This is the most important step. Without personalized (familiar) music, the whole system is less valuable. My Mom mailed me my Dad's favorite CDs. Place your music in the ~/dqmusicbox/music folder. Use one subfolder per album as you see below. DQMusicBox plays the music in alphabetical order by folder name and in alphabetical order by track name/number. MP3, FLAC, and AAC/MP4/iTunes files are supported. For maximum happiness, remove my test tracks (once everything is working). I use [WinSCP](#) to transfer music files to DQMusicBox:



6.6 Add shell script to automatically start the musicbox

```
sudo cp ~/dqmusicbox/bin/dqmusicbox.sh /etc/init.d
sudo chmod 755 /etc/init.d/dqmusicbox.sh
sudo update-rc.d dqmusicbox.sh defaults
```

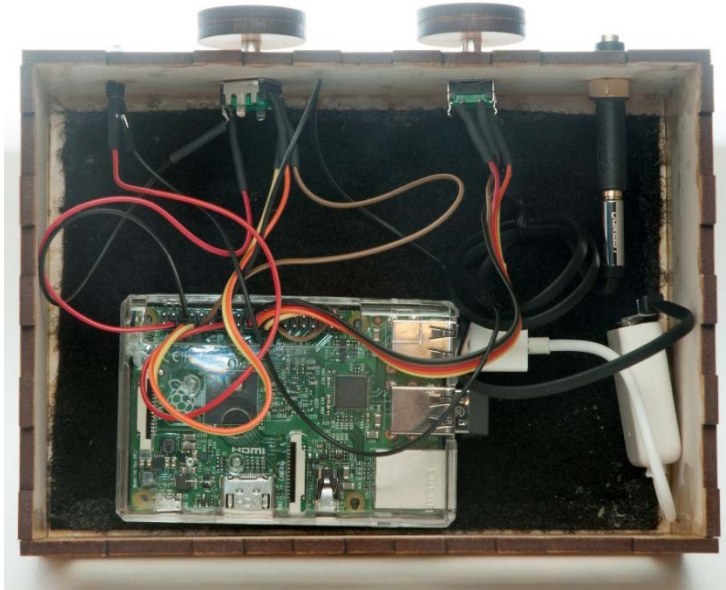
For more information startup scripts, see Stephen Christopher Phillips' [terrific page](#).

6.7 Reboot

```
sudo reboot
```

7 Put it all together

This is what you should get when you put it all together:



I don't think you need all the steps listed out. But here are a few tips for putting it all together:

1. Mount the front panel items before gluing - the soldered rotary encoders (already soldered to jumper wires), the indicator LED, and the headphone jack.
2. Solder the LED to the jumper wires.
3. Attach velcro to the bottom before gluing.
4. Glue the bottom and sides of the wood case. Put the top on, but don't glue the top on.
5. Use painter's tape (clamps would be better) to hold the box together while the glue dries.
6. Use velcro to hold the raspberry pi plastic case in place.
7. Use velcro to hold the thick end of the USB audio adapter in place.
8. I obsessed a bit about the knobs, so you have some choices:
 - a. Use the wood knobs from the case cut. If desired, glue together a few of the knob circles to make deeper knobs. These knobs are pictured above and in the main case phone on page 1. Note that you may need to file these knobs a bit before they will fit on the shafts of the rotary encoders.
 - b. [Eagle knobs from mouser](#). Black rubber. Simple, inexpensive, but smaller than I would like them to be.
 - c. 3D print some knobs, such as [this one from thingiverse](#). But note that I had to print a few before I got the size just right (plan +15%). These are the knobs on the dqmusicbox in the photo with my Dad on page 1.

8 Test cases

Congratulations on assembling everything. Now it is time to test. You will find it helpful to be looking at the log file (`tail -f /var/log/dqmusicbox.log`) for some of these tests.

Name	Description & Expectation
Light 1min after power on	The indicator light turns on when DQMusicBox is ready to play music, which is generally about 1 minute after power on.
Start song	Turning either of the knobs will start music playing.
Change song	Turn the songs knob. If you go forward and backward through the song list as expected, then all is well. If the songs list goes backward when you expect it to go forward, then you may be gotten a rotary encoder that reverses the clockwise and anti-clockwise pins. The fix is easy – just swap the orange and yellow jumper wires i.e. swap the pins that these jumper wires plug into.
Change volume	Turn the volume knob. If the volume goes up and down as expected, then all is well. If the volume goes down when you expect it to go up, then you may be gotten a rotary encoder that reverses the clockwise and anti-clockwise pins. The fix is easy – just swap the orange and yellow jumper wires i.e. swap the pins that these jumper wires plug into.
Pause	Tap the volume knob. Tap the songs knob. Both should work.
Shut down	Long hold (15-30 seconds) on the volume knob.
Reboot	Long hold (15-30 seconds) on the songs knob.
SSH	Connecting via PuTTY should work.
SFTP	Transferring files via WinSCP or similar should work.