Macintosh HD:Users:GleasonK:Desktop:Screen Shot 2014-05-15 at 11.41.30 PM.pngMacintosh HD:Users:GleasonK:Desktop:Screen Shot 2014-05-15 at 11.35.11 PM.pngMacintosh HD:Users:GleasonK:Desktop:Screen Shot 2014-05-15 at 11.37.04 PM.pngMacintosh HD:Users:GleasonK:Desktop:Screen Shot 2014-05-15 at 11.39.14 PM.png

**First, links to everything:**

**Download PiFm**

<http://www.icrobotics.co.uk/wiki/index.php/Turning_the_Raspberry_Pi_Into_an_FM_Transmitter>

**or** <http://www.instructables.com/id/Raspberry-Pi-Radio-Transmitter/?ALLSTEPS>

**Play mp3 on PiFm**

<https://docs.google.com/document/d/1URn_9QpnP9CjUq9fpjuMdpL6svTr8hoGThmilHePV5g/edit>

**Screen Commands**

<http://www.tecmint.com/screen-command-examples-to-manage-linux-terminals/>

**Make Linux Shell Script**

<http://www.linfo.org/create_shell_1.html>

Get to folder you want. Mine is ~/Applications/PiFM/

wget http://omattos.com/pifm.tar.gz # Download PiFm

tar –zxvf pifm.tar.gz # Decompress file

sudo ./pifm sound.wav 100.0 # Test to make sure all worked out

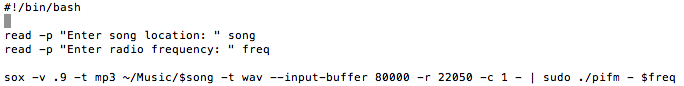
NOTE: To make your signal better, attach a wire to GPIO 4. This antenna makes it very powerful.

🡪 GPIO 4 is the 4th pin from the SD Card on the inside (The row closer to SD Card).

Add songs to ~/Music/ using FTP like FileZilla or other. PiFm can only play WAV to bypass that..

sudo apt-get install sox libsox-fmt-all # Download all mp3-convert files

sox -v .9 -t mp3 SONG -t wav --input-buffer 80000 -r 22050 -c 1 - | sudo ./pifm – FREQUENCY # Long command to play the mp3 SONG on FREQ mHz.



**Quick look at my directory layout, made Applications and Music folder in ~ . Refer to this later.**

Now, instructions so this can be pain free, hopefully.

sudo apt-get update

sudo apt-get install screen

🡪 Screen allows you to SSH into Pi then make a screen (terminal shell) that you can detach from so when you turn off computer the script you had going doesn’t terminate

🡪 Useful commands:

🡪🡪 screen (makes screen) 🡪 Ctrl-A + D (detach screen so you can log out)

🡪🡪 screen –list (lists all screens) 🡪 screen –r [name] (name from –list, this reattaches)

## Successfully plays: ~/Music/BoB/Airplanes.mp3

## Plays all songs by BoB

## Refer to vim file if lost

## Radio FM 99.0 mHz

That will play that one song, all is well. But that is a pain to type out and to have to do that every time sucks. **So how about this**:

**Even better:**

## Plays all songs by

## All artists

**Last example, my favorite.** You can play around with combinations to find files or artists.

**The lazy search for song:**

**ENJOY**

**To avoid having to type this command every time I made an executable:**

vim mp3Play # sudo apt-get install vim if you have not yet

######## Following lines are the vim file – mp3Play ##########

#!/bin/bash

read -p "Enter song location: " song

read -p "Enter radio frequency: " freq

sox -v .9 -t mp3 ~/Music/$song -t wav --input-buffer 80000 -r 22050 -c 1 - | sudo ./pifm - $freq

######## File end ##########

Essentially, I just took the long command and used variable inserts for the song based on user prompts.

**This script assumes all music is in ~/Music folder**. I have artist folders inside that, you don’t have to, but it is nice to stay organized.

**Read** gets user input. Also, note how variables are used. After made in read, called with $

However, trying to execute: ./mp3Play will not work because it needs permissions so run

chmod 755 morning # Allows read/write/execute permissions

**Now here’s a cool part. You could guess I’d be able to call a song like this:**