

Raspberry Pi 2 Model B FM Radio Transmitter (Basic notes on Software and Antenna)

```
# This document has the requirements to broadcast / transmit from a RPi 2 to ANY
# radio reciever (Audio channels usually) within the frequency ranges of 76 to -
# 108 MHz
#
# NOTE - theres a FEW software packages out there for this FM transmitter RPi -
# project
# They don't all work and I tried them ALL (August 2016)
# Below is the way I got it working exactly in order of what to type
# I used a Raspberry Pi2 Model B - Running Raspbian Jessie updated regularly
# This software is NOT my own work. It is called "PiFmRds" and can be found on -
# Github
#
#
# As I mess round with it I will update my file for this software, and my -
# Ultimate goal is to get this down to CB UK 40 channel frequencies and talk -
# Using a voice synthesizer or something. 27 MHz to 29 MHz approx
#
# NEWBIES - wherever you see this symbol ($-) you type what follows - BLUE
# colour writing.
# It represents the command terminal cursor mark.
#
# Ben Woodfield. Hardware salvager, CB'er, Studying for Ham Radio License,
# Software writer, Python Programmer, Electronics nut, High Voltage Headcase
# CB Handle: Apprentice / Location: South Wales UK
```

Below are the installation and download commands - in order

```
$-sudo apt-get install sox
$-sudo apt-get install libsndfile1-dev
$-git clone https://github.com/ChristopheJacquet/PiFmRds.git
$-cd PiFmRds/src
$-make clean      # NOTE I had an error on this command. Just skipped it, no
issues yet.
$-make
```

```
# Installation Finished. At this point if you had NO other errors youre good to
go.
```

```
# Now to open the new program (it runs from the command terminal....no GUI)
$-sudo ./pi_fm_rds
```


X = GPIO Pin 4

○	○
○	○
○	○
X	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○
○	○

X = GPIO Pin 4.....4th down from top on the left column...just google a pic...

REMEMBER the pin numbers and GPIO pin numbers are different. Some genius is a complete idiot.

So a manual may say the pin numbers in order of their placement but GPIO numbers are based on their function

Get a reliable chart for YOUR MODEL OF Rpi

The longer your wire antenna the stronger you will transmit a signal. If you DONT want to break the law where you live or interrupt life support - machines...shorten the antenna so ONLY YOU can receive it on those frequencies.

Shorter antenna = Shorter range

Antenna Designs:

Now I got a 20cm jumper wire and attached it to GPIO4. Then connected about 50cm of speaker wire to the jumper and taped the end up vertically like an antenna should be. This works great. I got out almost 100 meters outdoors with my Pi transmitting from indoors

No antenna connected will get a weak signal to a stereo on the same table depending on the antenna positioning of the stereo

It WILL work with just the GPIO pin but an antenna is better

Now I haven't tried this yet but I have seen Dipole antennas being made for this, utilising a power wire as the Ground / lower pole of the antenna. If you're familiar with CB and Ham you know what a dipole is already and this will

come naturally

If you're not...it is 2 poles (di-pole) one up one down. Ground to the lower pole which is the shielded or braided part of the coax wire, Inner to the top (Based on using COAX cable. The better you buy the thicker the inner braiding/shielding and the better the ground, and reception/transmission shielding from interference)

You only need a wire. Up to a meter long touching GPIO4 (TRY to use a proper jumper wire to prevent ANY shorts to other pins. You could fry your pi if you short some of them)

I have a breakout kit...it is a cable and connector that clips onto my whole row of GPIO pins and then connects to a breadboard. I can easily and safely put jumper wires into this without risk of shorting anything! I connect a jumper wire to a meter of speaker wire, and just tape it to something vertical (the wall for me)