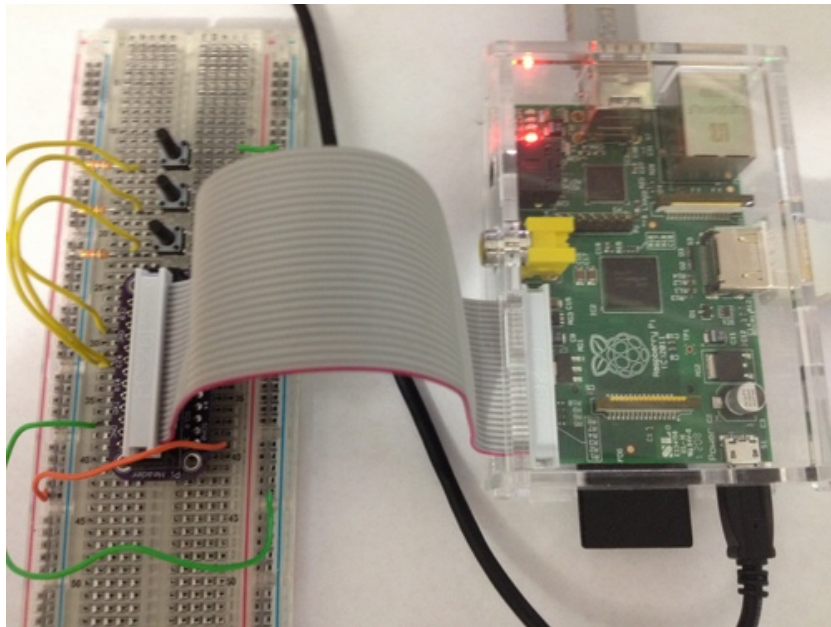


## Playing sounds and using buttons with Raspberry Pi

Created by Mikey Sklar

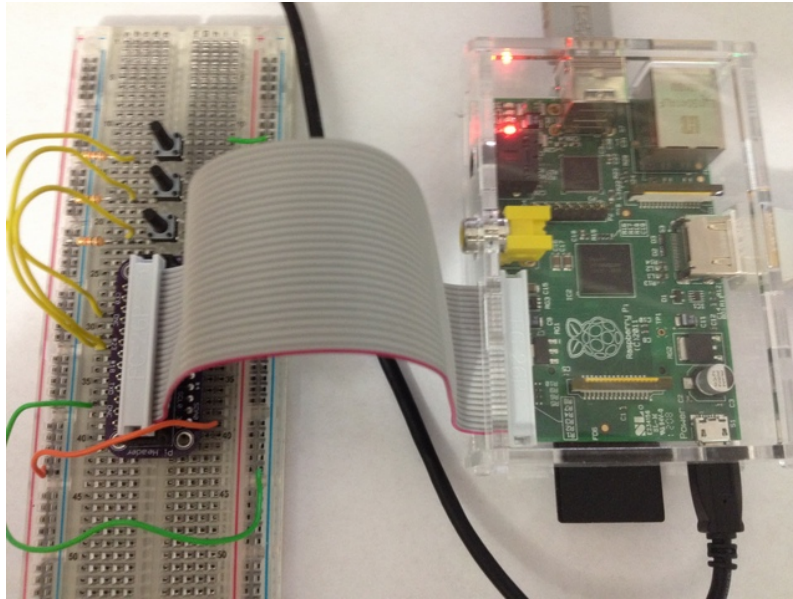


## Guide Contents

Guide Contents	2
Overview	3
Install Audio	4
Install Python Module RPi.GPIO	6
Bread Board Setup for Input Buttons	7
Code	9

## Overview

---



One of the great things about the Raspberry Pi is how everyone starts with same piece of gear. Since the audio hardware is identical on every unit it is trivial to load the drivers and play mp3 files.

This guide describes how to connect input buttons and play audio files using a Raspberry Pi. We make use of the Adafruit's Pi Cobbler Breakout Kit and the python module RPi.GPIO. If you have not already used the raspberry pi as a input device this guide will show you how to wire the pull-down resistors, GPIO pins and buttons.

## Install Audio

With the Pi connected to the Internet and SSH'ed in ([see our previous tutorial \(http://adafru.it/aj5\)](http://adafru.it/aj5)) install the alsa audio drivers and MP3 Player

```
$ sudo apt-get install alsa-utils
$ sudo apt-get install mpg321
```

```
pi@raspberrypi:~$ sudo apt-get install alsa-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  alsa-base linux-sound-base lsdf
Suggested packages:
  apod alsa-oss oss-compat
The following NEW packages will be installed:
  alsa-base alsa-utils linux-sound-base lsdf
0 upgraded, 4 newly installed, 0 to remove and 42 not upgraded.
Need to get 1,747 kB of archives.
After this operation, 3,138 kB of additional disk space will be used.
Do you want to continue [Y/n]?
Get:1 http://ftp.uk.debian.org/debian/ squeeze/main lsdf armel 4.81.dfsg.1-1 [282 kB]
Get:2 http://ftp.uk.debian.org/debian/ squeeze/main linux-sound-base all 1.0.23+dfsg-2 [29.0 kB]
Get:3 http://ftp.uk.debian.org/debian/ squeeze/main alsa-base all 1.0.23+dfsg-2 [313 kB]
Get:4 http://ftp.uk.debian.org/debian/ squeeze/main alsa-utils armel 1.0.23-3 [1,124 kB]
Fetched 1,747 kB in 5min 34s (5,228 B/s)
Preconfiguring packages ...
Selecting previously deselected package lsdf.
(Reading database ... 47664 files and directories currently installed.)
Unpacking lsdf (from .../lsdf_4.81.dfsg.1-1_armel.deb) ...
Selecting previously deselected package linux-sound-base.
Unpacking linux-sound-base (from .../linux-sound-base_1.0.23+dfsg-2_all.deb) ...
Selecting previously deselected package alsa-base.
Unpacking alsa-base (from .../alsa-base_1.0.23+dfsg-2_all.deb) ...
Selecting previously deselected package alsa-utils.
Unpacking alsa-utils (from .../alsa-utils_1.0.23-3_armel.deb) ...
Processing triggers for man-db ...
Setting up lsdf (4.81.dfsg.1-1) ...
Setting up linux-sound-base (1.0.23+dfsg-2) ...
Setting up alsa-base (1.0.23+dfsg-2) ...
Setting up alsa-utils (1.0.23-3) ...
pi@raspberrypi:~$
```

```
pi@raspberrypi:~$ sudo apt-get install mpg321
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  libao-common libao4 libaudio-scrobbler-perl libconfig-inifiles-perl
Suggested packages:
  libaudio2 libpulse0 libdata-dump-perl libcrypt-ssleay-perl libio-socket-ssl-perl
The following NEW packages will be installed:
  libao-common libao4 libaudio-scrobbler-perl libconfig-inifiles-perl
0 upgraded, 14 newly installed, 0 to remove and 42 not upgraded.
Need to get 1,181 kB of archives.
After this operation, 3,768 kB of additional disk space will be used.
Do you want to continue [Y/n]?
```

Reboot the Pi (% **reboot**) and when it comes back up, load Sound Drivers and Setup for 3.5mm Jack Output

```
$ sudo modprobe snd_bcm2835
$ sudo amixer cset numid=3 1
```

```
pi@raspberrypi:~$ sudo modprobe snd_bcm2835
pi@raspberrypi:~$ sudo amixer cset numid=3 1
numid=3,iface=MIXER,name='PCM Playback Route'
; type=INTEGER,access=rw-----,values=1,min=0,max=3,step=0
: values=1
```

---

## Install Python Module RPi.GPIO

---

The RPi.GPIO python module offers easy access to the general purpose IO pins on the Raspberry Pi.

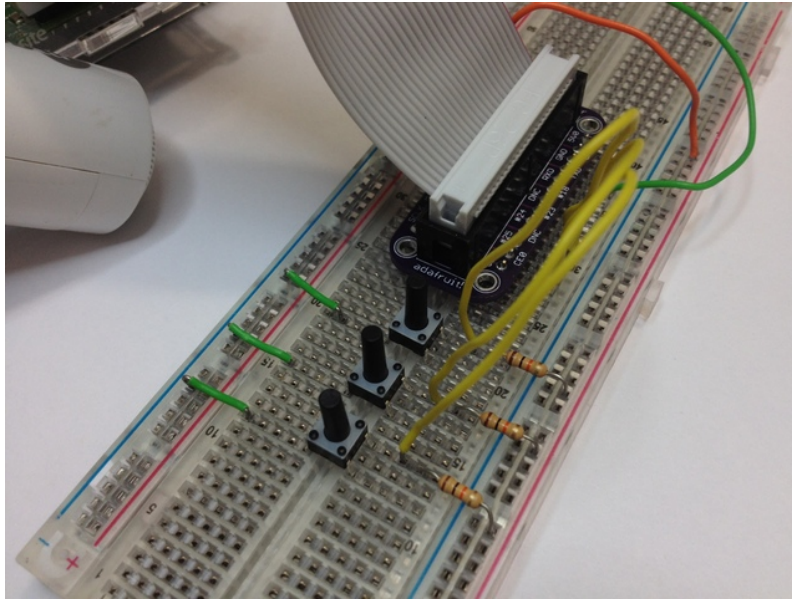
To get the latest version of this, take a little diversion to follow the instructions in this Adafruit Lesson: <http://learn.adafruit.com/adafruits-raspberry-pi-lesson-4-gpio-setup> (<http://adafru.it/aTH>)

If you are an advanced user, you can probably skip the lesson above and issue the following commands in a Terminal window.

```
$sudo apt-get update
$sudo apt-get install python-dev
$sudo apt-get install python-rpi.gpio
```

## Bread Board Setup for Input Buttons

---

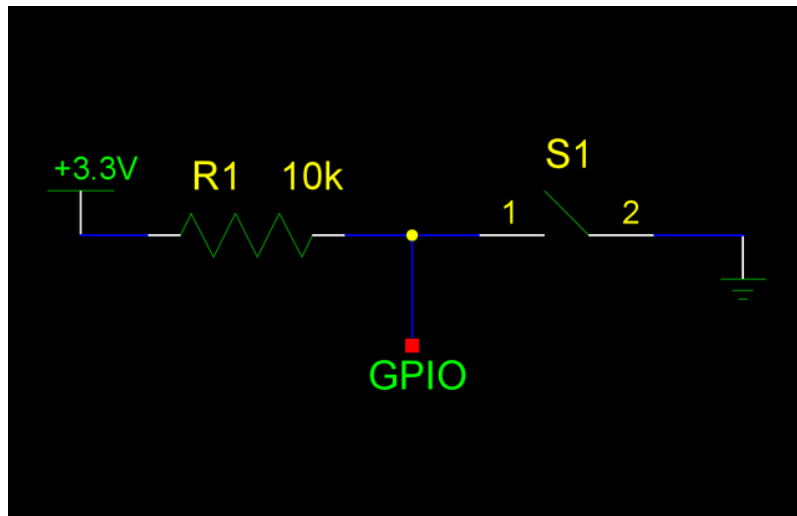


Important things to note:

- The [Adafruit Pi Cobbler Breakout Kit \(http://adafru.it/914\)](http://adafru.it/914) is being used
- The [Adafruit Clear Full sized breadboard \(http://adafru.it/239\)](http://adafru.it/239) is being used
- (3) 10k pull-up resistors
- (3) [Momentary push-button switches \(http://adafru.it/367\)](http://adafru.it/367)
- GPIO pins 23, 24 and 25
- Cobbler 3.3v rail is tied to positive breadboard
- Cobbler gnd rail is tied to negative breadboard

Each button connection looks like:

3.3v --> 10k Pull-up Resistor --> GPIO --> Button --> GND





## Code

- Insert this code into a file named **raspi-audio-button.py**
- Download or copy 3 mp3 files to the Pi and place them into the same directory as the **raspi-audio-button.py** script. Replace the file names with the ones in the code
- Make the file executable with **chmod**

```
#!/usr/bin/env python

from time import sleep
import os
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
GPIO.setup(23, GPIO.IN)
GPIO.setup(24, GPIO.IN)
GPIO.setup(25, GPIO.IN)

while True:
    if ( GPIO.input(23) == False ):
        os.system('mpg321 binary-language-moisture-evaporators.mp3 &')
    if ( GPIO.input(24) == False ):
        os.system('mpg321 power-converters.mp3 &')
    if ( GPIO.input(25) == False ):
        os.system('mpg321 vader.mp3 &')
    sleep(0.1);
```

```
#!/usr/bin/env python
from time import sleep
import os
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BCM)
GPIO.setup(23, GPIO.IN)
GPIO.setup(24, GPIO.IN)
GPIO.setup(25, GPIO.IN)

while True:
    if ( GPIO.input(23) == False ):
        os.system('mpg321 binary-language-moisture-evaporators.mp3 &')
    if ( GPIO.input(24) == False ):
        os.system('mpg321 power-converters.mp3 &')
    if ( GPIO.input(25) == False ):
        os.system('mpg321 vader.mp3 &')
    sleep(1);
```

```
$ chmod +x raspi-audio-button.py
```

Run the python program as an administrator (with **sudo**). Press the button keys to hear the mp3 files play. Make sure you have speakers or headphones hooked up to the 3.5mm jack.

```
$ sudo python raspi-audio-button.py
```

