



## AdaBox 004

Created by Tyler Cooper



Last updated on 2018-08-22 03:59:52 PM UTC

## Guide Contents

Guide Contents	2
Introduction	3
Hi there!	3
Who is this for?	3
Who isn't this for?	3
Who are you?	3
Want to buy past AdaBoxes?	3
AdaBox001 - Welcome to the Feather Ecosystem	4
AdaBox002 – Making Things Move	4
AdaBox003 – The World of IoT – Curated by Digi-Key	4
AdaBox004 – Making Things Dance	4
AdaBox005 – Break for Pi	5
Unboxing AdaBox 004	6
Kit Contents	6
Bonus parts (for subscribers only!)	7
Getting started with your new AdaBox	8
The HUZZAH ESP8266 Feather	9
The MusicMaker FeatherWing	11
Building your music player	13
Loading music onto the SD card	16
USB MicroSD Card Reader/Writer - microSD / microSDHC / microSDXC	16
Arduino IDE and Installing New Libraries	18
Playing your first mp3	20
Streaming Internet Radio	21
Adding a volume knob	24
Adding a power switch	26
IR Remote and Receiver	27
AdaBox 004 Projects	32
Project 1: WiFi Music Alert Box	33
Project 2: 3D Printed Boomy the Boombox	34
Project 3: The Haunted Radio	35
AdaBox Upgrade!	35
Bonus Custom BartleBeats AdaBox 004 Soundtrack!	36
Share your MusicMaker box designs and get help	37

## Introduction

---

### Hi there!

---

If you're looking to subscribe to AdaBox, click here! (<https://adafru.it/tNC>)

If you're here, it's because you were given the gift of electronics with an AdaBox! You are a beginner who is getting started with your AdaBox or you just want to relive what it's like being a beginner at electronics again. But most of all, you want to learn how to build and make stuff with electronics! (If, rather than learn electronics, you'd like to look at pictures of cats instead, please check <https://www.adafruit.com/galleries/cats-of-engineering> (<https://adafru.it/oAd>))

And, you're in luck: there's *never* been a better time.

Gone are the days where you need thousands of dollars of equipment and lots physics/math background. Nowadays, if you want to learn to work with electronics, you can jump right in for \$100 or less, and any sort of computer. And we're talking about learning *a lot* of electronics - from the basics of analog to the complexities of firmware. With a good pack of parts, you can build a base of knowledge that will take you from your first blinking LED to someone who can start prototyping and inventing custom products.

### Who is this for?

---

Anyone with a computer they can install software on, an Arduino or compatible and the ability to type and click a mouse. That's pretty much the minimum. Remember, this guide is specifically for people who have purchased or received an AdaBox subscription!

**You don't need to know a lot of physics or math**, and just like an Art Degree isn't required for making art and being creative, **you *don't* need to have a computer science degree**. It helps if you're comfortable using computers but that's a skill most people pick up through life.

**If you know how to program already - great! If not, don't worry, we'll teach you enough to be dangerous.**

### Who isn't this for?

---

While you can follow along without an AdaBox, it will not make as much sense unless you have *all* of the components and more which either came as a gift or purchased yourself - remember, the goal is helping beginners!

If you're an expert, please visit our hundreds other tutorials and jump right in at [learn.adafruit.com](https://adafru.it/rdw) (<https://adafru.it/rdw>)

### Who are you?

---

Great question. This is me:

I'm Ladyada, and I love to teach people how to build stuff and how they can be creative with technology.

So, are you ready?

Let's do this thing!

### Want to buy past AdaBoxes?

---



AdaBox001 - Welcome to the Feather Ecosystem

\$70.00  
IN STOCK

ADD TO CART



AdaBox002 – Making Things Move

\$79.95  
IN STOCK

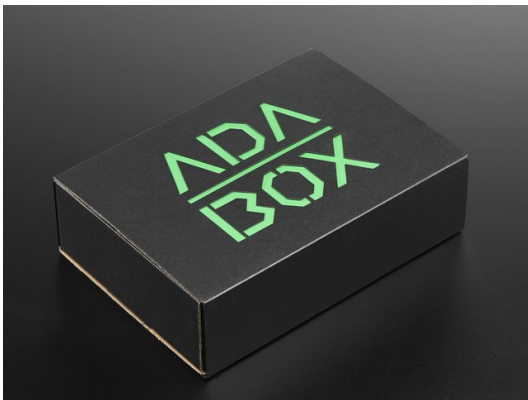
ADD TO CART



AdaBox003 – The World of IoT – Curated by Digi-Key

\$79.95  
IN STOCK

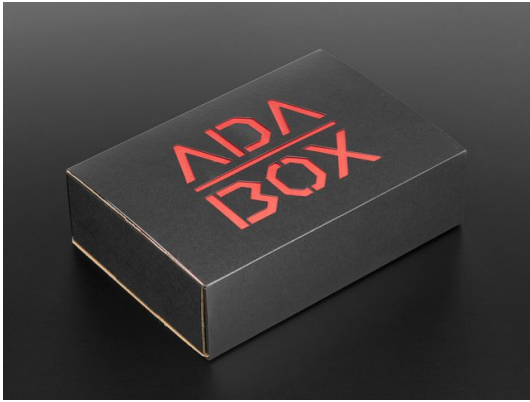
ADD TO CART



AdaBox004 – Making Things Dance

\$79.95  
OUT OF STOCK

OUT OF STOCK



AdaBox005 – Break for Pi

\$69.95  
IN STOCK

ADD TO CART

## Unboxing AdaBox 004

Sit down, strap in, and pump your Feather ecosystem up to 11. The last of our Feather-focused AdaBoxes has all the ingredients for a remote controlled music player, a Zen-inducing meditation machine, or your very own Radio Free Adafruit that streams tunes through the internet.

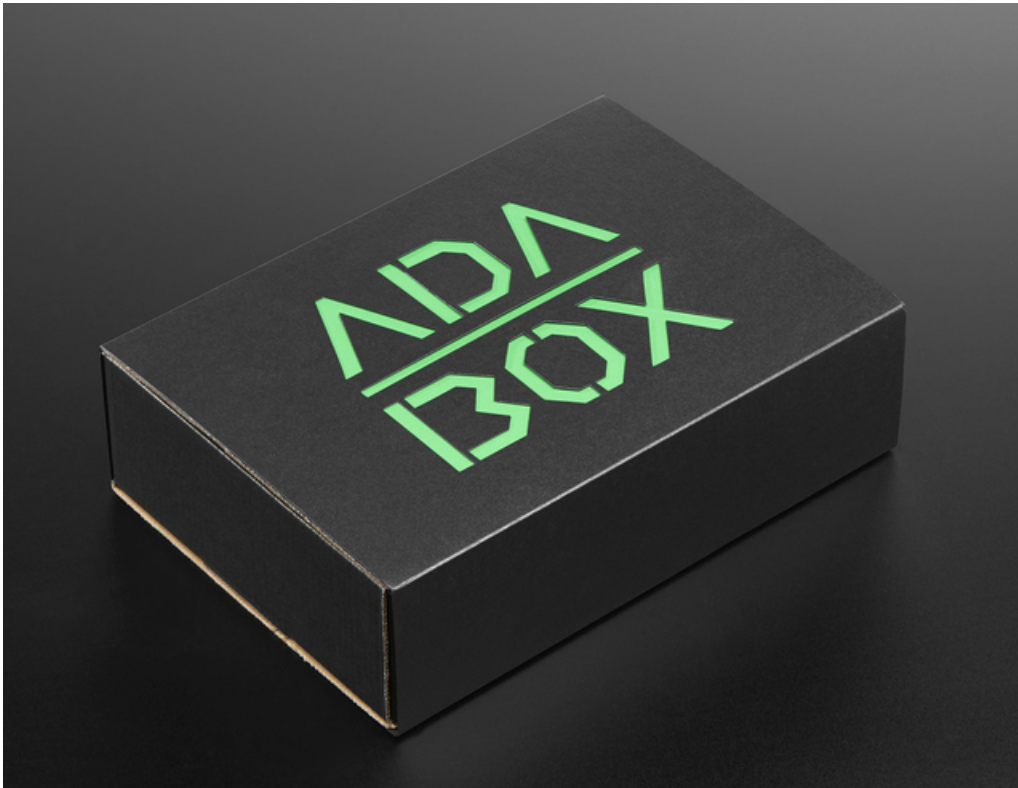
In honor of one year of AdaBoxes, Adafruit asks that you savor AdaBox004 and create whatever your ears can imagine.



## Kit Contents

- [Feather HUZZAH with ESP8266 WiFi - pre-soldered with Feather Stacking Headers \(https://adafru.it/xcj\)](https://adafru.it/xcj) - our take on an 'all-in-one' ESP8266 WiFi development board with built in USB and battery charging. Its an ESP8266 WiFi module with all the extras you need, ready to rock!
- [MusicMaker FeatherWing with Amplifier - pre-soldered \(https://adafru.it/w0f\)](https://adafru.it/w0f) - It's a fun-size version of our [Music Maker shield for Arduino \(https://adafru.it/tbM\)](https://adafru.it/tbM), and similar to [our other Music Maker FeatherWing \(http://adafru.it/3357\)](http://adafru.it/3357), but this one is even better - **with an on-board 3 Watt amplifier!**
- 3W 4 ohm Speaker with wires stripped and tinned
- [Mini Remote Control \(https://adafru.it/xck\)](https://adafru.it/xck) - This little remote control would be handy for controlling a robot or other project from across the room. It has 21 buttons and a layout we thought was handy: directional buttons and number entry buttons.
- [IR Sensor \(https://adafru.it/xcl\)](https://adafru.it/xcl) - IR sensor tuned to 38KHz, perfect for receiving commands from a TV remote control. Runs at 3V to 5V so it's great for any microcontroller.
- [Premium Male/Male Jumper Wires \(https://adafru.it/sd3\)](https://adafru.it/sd3) - Handy for making wire harnesses or jumpering between headers. Essential for most electronics projects.

- [USB Cable - A/MicroB \(https://adafru.it/iia\)](https://adafru.it/iia)
- [Half Size Breadboard \(https://adafru.it/keP\)](https://adafru.it/keP) - This is a cute half size breadboard, good for small projects. Also essential for most electronics projects.
- [10K Breadboard Potentiometer \(https://adafru.it/tba\)](https://adafru.it/tba) - use this to control your volume and more.
- [SPDT Switch \(https://adafru.it/drN\)](https://adafru.it/drN) - perfect for adding an on/off switch to your music box
- [5V1A USB Power Supply \(https://adafru.it/iia\)](https://adafru.it/iia) - power your little music box with this, or connect batteries to take your music box on the go.
- AdaBox004 component bag - This is really just a single resistor :) We wanted to keep this project simple and easy to make!
- A blank box - with which imagination is your only limitation



## Bonus parts (for subscribers only!)

---

- Digikey's Corner - Our good friends at Digikey threw in a wonderful micro SD card for your listening pleasure!
- Adafruit Collectible Pin - Meet BOOMY the loveable boombox. BOOMY's one of our favorite companions whenever we start putting together a project since the songs that come out of him helps keep our projects moving and grooving. His favorite artists are Young Thug, deep album Yazoo, and the genre of cantu a tenore - a traditional Sardinian folk music!
- Signed copy of Make Magazine - have you ever signed 3500 autographs? Because ladyada has!

## Getting started with your new AdaBox

---

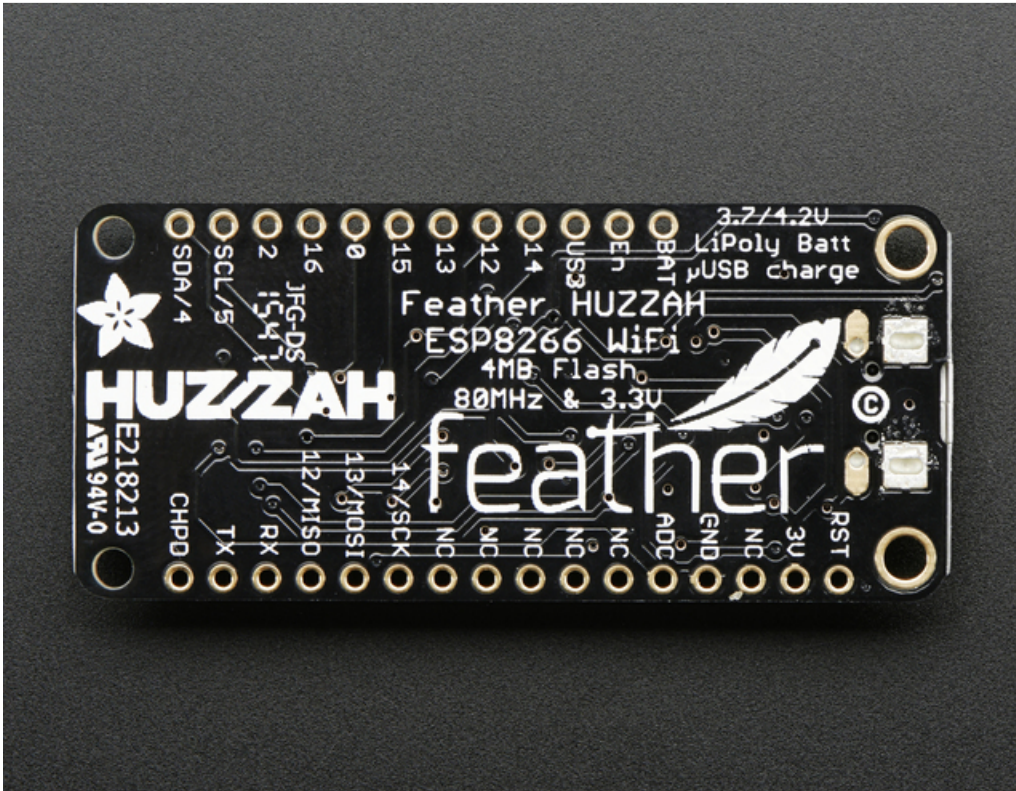


The first thing we are going to do is get your new HUZDAH Feather and MusicMaker FeatherWing working with your computer and the Arduino IDE. Then we will show you how to load up music onto the included SD card, and start playing that music through the included speaker.

From there we can take things to the next level by streaming music over the internet, using the IR remote to control music, connecting to Adafruit IO and making a WiFi Music Alert Box, and building a sweet 3D printed Boomy the Boombox speaker!



## The HUZZAH ESP8266 Feather



If you have received our previous AdaBoxes, you should be pretty familiar with the HUZZAH ESP8266 Feather. This is the board that you will plug your MusicMaker FeatherWing into. This little board packs a big punch:

Feather is the new development board from Adafruit, and like its namesake it is thin, light, and lets you fly! We designed Feather to be a new standard for portable microcontroller cores.

This is the **Adafruit Feather HUZZAH ESP8266** - our take on an 'all-in-one' ESP8266 WiFi development board with built in USB and battery charging. Its an ESP8266 WiFi module with all the extras you need, ready to rock! [We have other boards in the Feather family, check'em out here \(https://adafru.it/jAQ\).](https://adafru.it/jAQ)

At the Feather HUZZAH's heart is an ESP8266 WiFi microcontroller clocked at 80 MHz and at 3.3V logic. This microcontroller contains a Tensilica chip core as well as a full WiFi stack. You can program the microcontroller using the Arduino IDE for an easy-to-run Internet of Things core. We wired up a USB-Serial chip that an upload code at a blistering 921600 baud for fast development time. It also has auto-reset so no noodling with pins and reset button pressings.

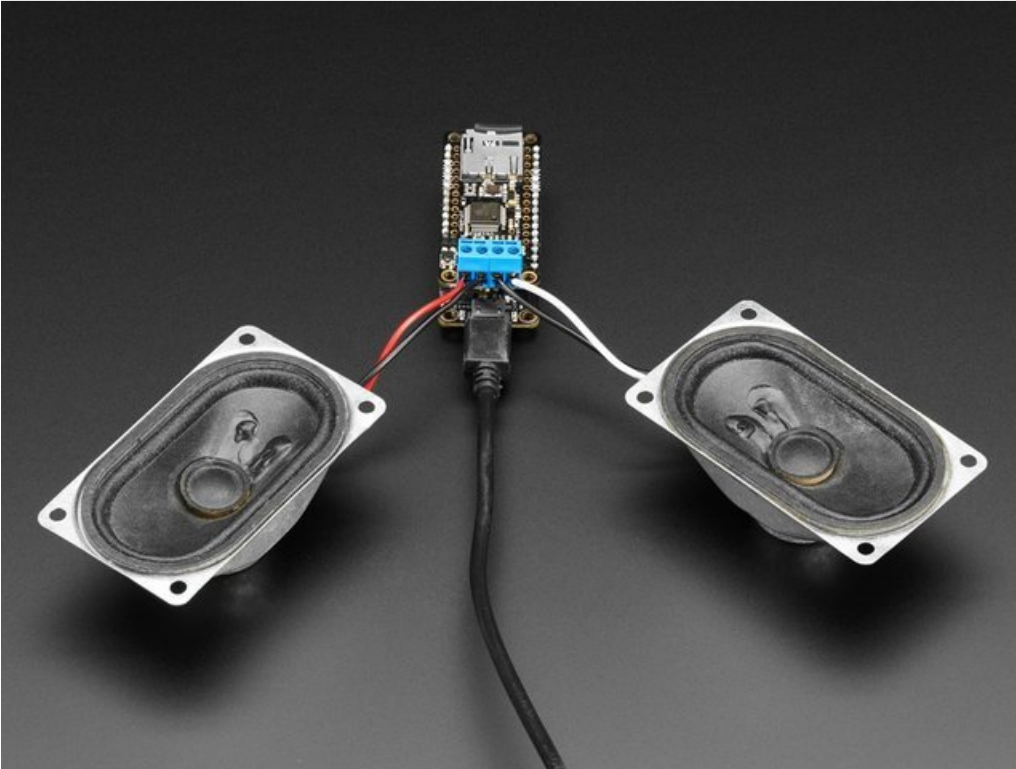
To make it easy to use for portable projects, we added a connector for any of our 3.7V Lithium polymer batteries and built in battery charging. You don't need a battery, it will run just fine straight from the micro USB connector. But, if you do have a battery, you can take it on the go, then plug in the USB to recharge. The Feather will automatically switch over to USB power when its available.

Head over to the HUZZAH Feather guide to learn more about this board and how to get it connected to the internet.

<https://adafru.it/kD3>

<https://adafru.it/kD3>

## The MusicMaker FeatherWing



Now that you have played around with your HUZAZH Feather, you are going to want to get acquainted with the star of the show, the MusicMaker FeatherWing:

Bend all audio files to your will with the **Adafruit Music Maker FeatherWing + Amplifier!** It's a fun-size version of our [Music Maker shield for Arduino \(https://adafru.it/tbM\)](https://adafru.it/tbM), and similar to [our other Music Maker FeatherWing \(http://adafru.it/3357\)](http://adafru.it/3357), but this one is even better - **with an on-board 3 Watt amplifier!** This powerful wing features the VS1053, an encoding/decoding (codec) chip that can decode a wide variety of audio formats such as MP3, AAC, Ogg Vorbis, WMA, MIDI, FLAC, WAV (PCM and ADPCM). You can do all sorts of stuff with the audio as well such as adjusting bass, treble, and volume digitally.

All this functionality is implemented in a light-weight SPI interface so that any Feather Board can play audio from an SD card. There's also a special MIDI mode that you can boot the chip into that will read 'classic' 31250Kbaud MIDI data from the UART TX pin and act like a synth/drum machine - there are dozens of built-in drum and sample effects!

What a great musical add-on to your Feather! That's why we spun up this super FeatherWing, perfect for use with any of our Feather Boards! Music playback testing working with ATmega32u4, ATSAMD M0, ESP8266, WICED Teensy 3, and nRF52 Feathers. MIDI playback works with all Feathers but you may need to remove the 'Wing during programming and also for USB-Serial Feathers like the ESP8266 and nRF52 you'll end up re-using the main Serial console for MIDI which can be confusing.

Here are some specs:

- Features the VS1053B codec chip - decodes Ogg Vorbis, MP3/MP2/MP1, MP4, AAC, WMA, FLAC, WAV/PCM, MIDI
- Stereo audio out to any 4ohm speakers (3W max) or 8 ohm speakers (1W max), class D output

- MicroSD card socket, for any FAT16/FAT32 formatted SD card from 64MB or greater
- Built in MIDI synth/drum machine with dozens of instruments

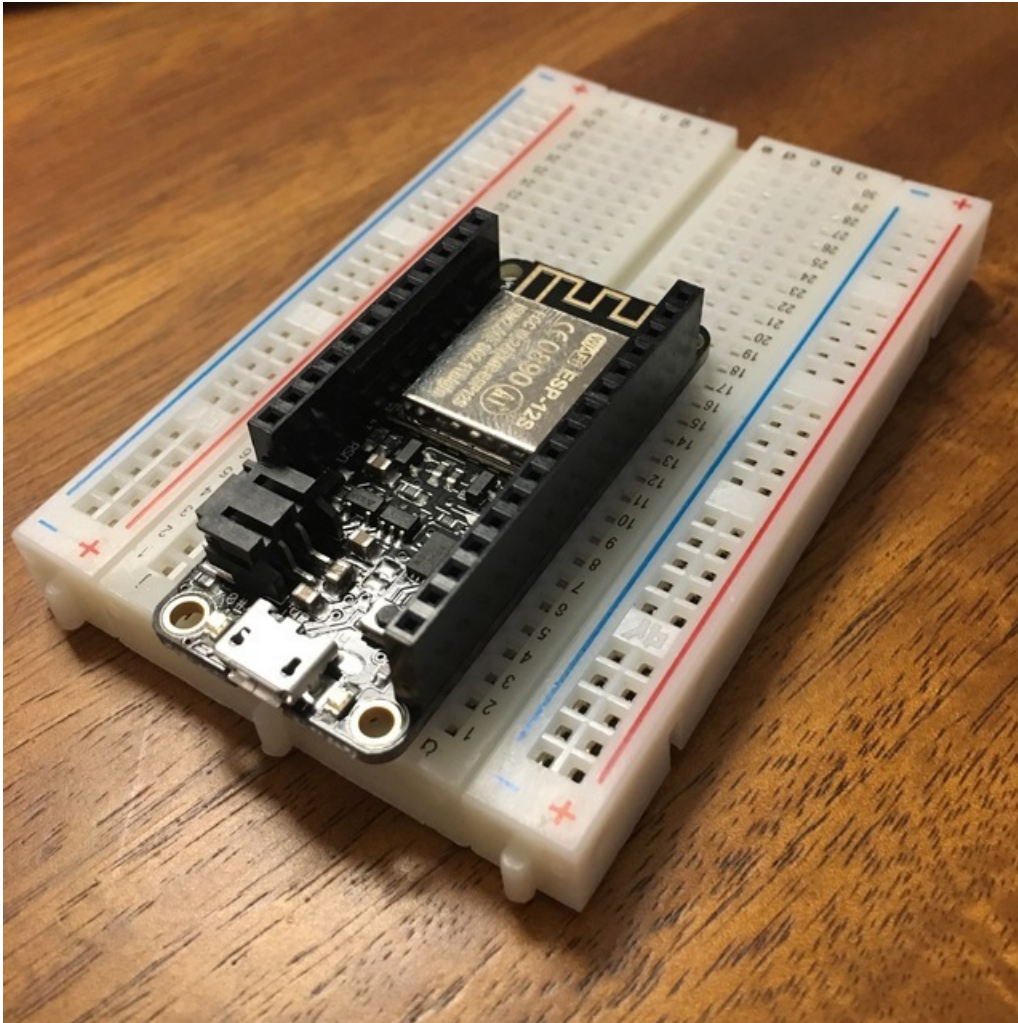
Start by heading over to the MusicMaker FeatherWing guide to learn how to get your MusicMaker FeatherWing to play music and make noises.

<https://adafru.it/xcm>

<https://adafru.it/xcm>

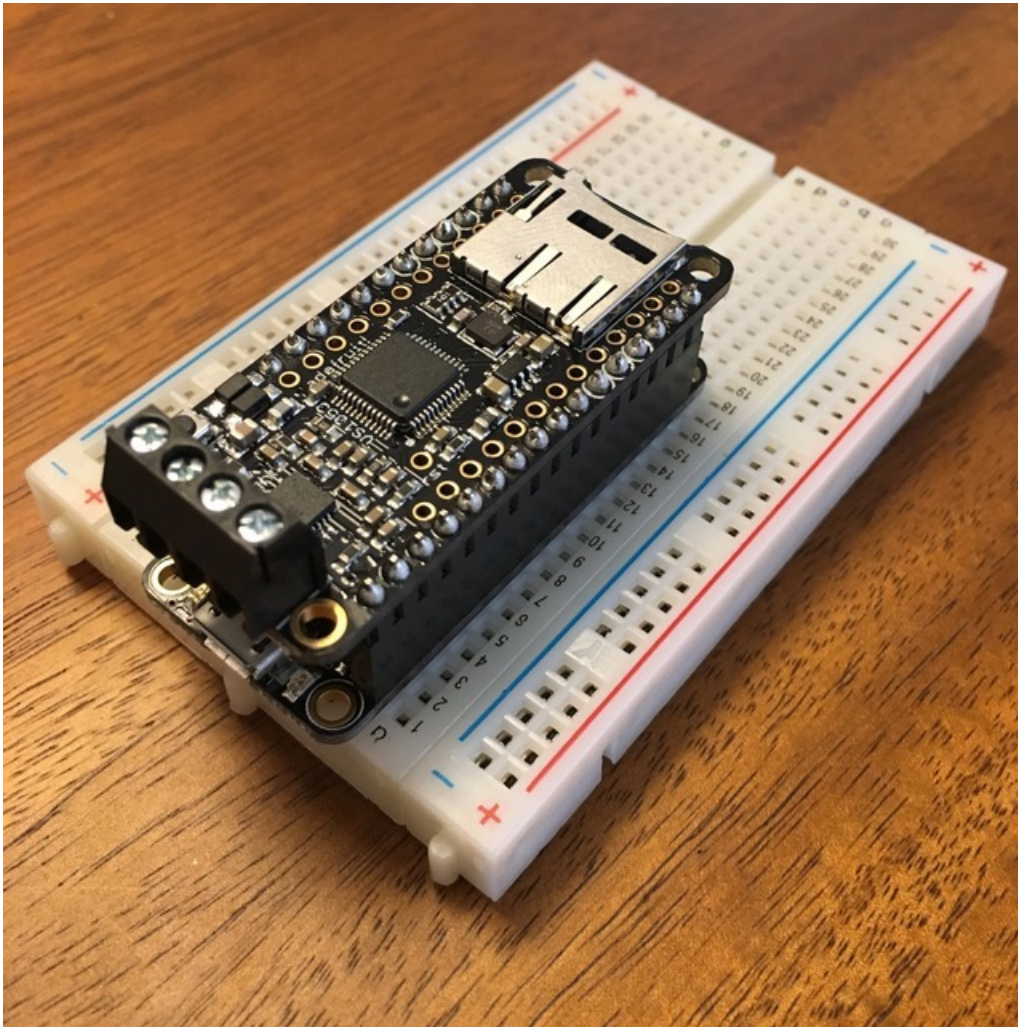
## Building your music player

We are going to build this music player right on the included breadboard. Start by plugging your HUZAZH Feather directly into the breadboard like this:



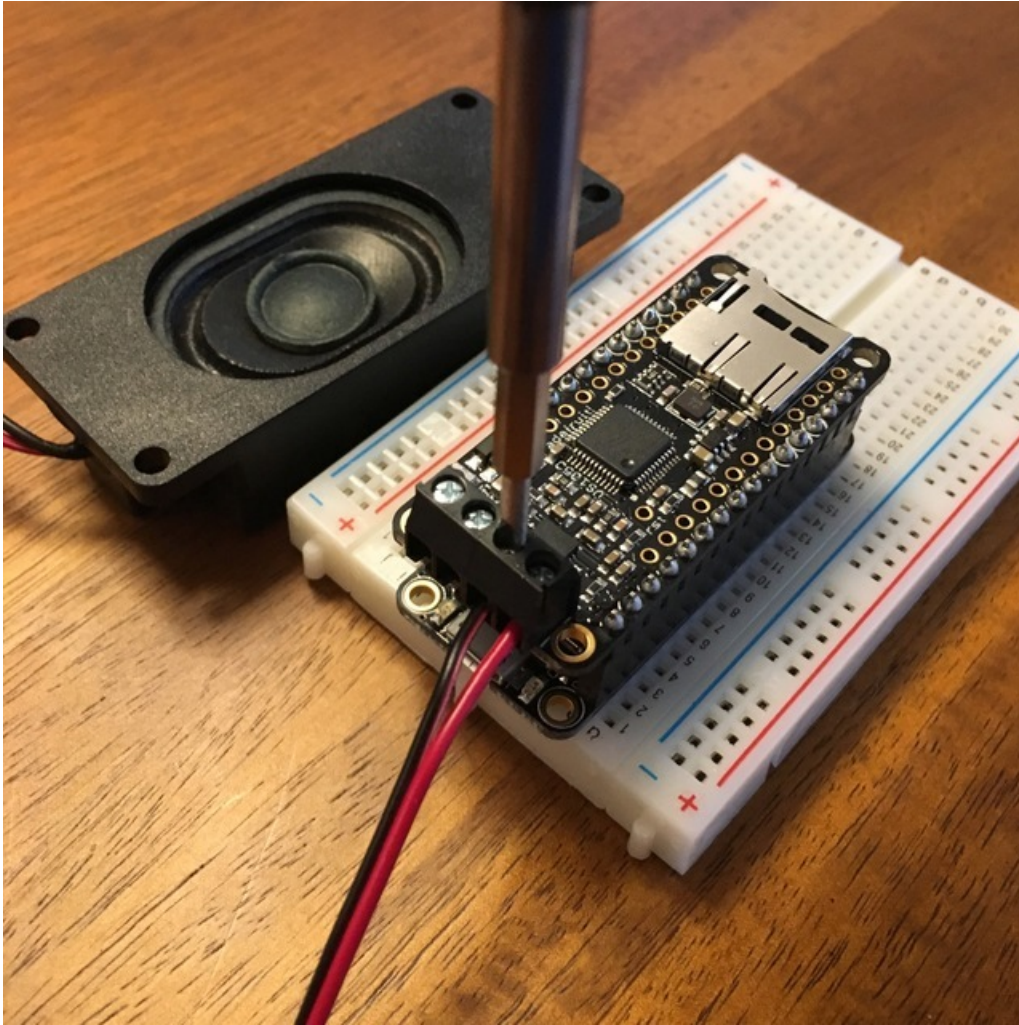
Then take your MusicMaker FeatherWing and plug it into the HUZAZH Feather:





Now, grab a small screwdriver and attach the speaker wires to the MusicMaker FeatherWing terminals. We are only going to use the one speaker, so it really doesn't matter which speaker channel you choose. You can put it in the right, or left side. Just make sure to connect the red (positive) wire to one of the outside terminals, and the black (negative) wire to one of the inside terminals.

In the picture below, you can see I attached the wires to the left channel with the red wire in the outside position.



Tighten the screws and we are ready to load some music onto the SD card.

## Loading music onto the SD card

For this step you will need an SD card reader. These are often built into computers, or you can pick one up from Adafruit below.



USB MicroSD Card Reader/Writer - microSD / microSDHC / microSDXC

\$5.95  
IN STOCK

ADD TO CART

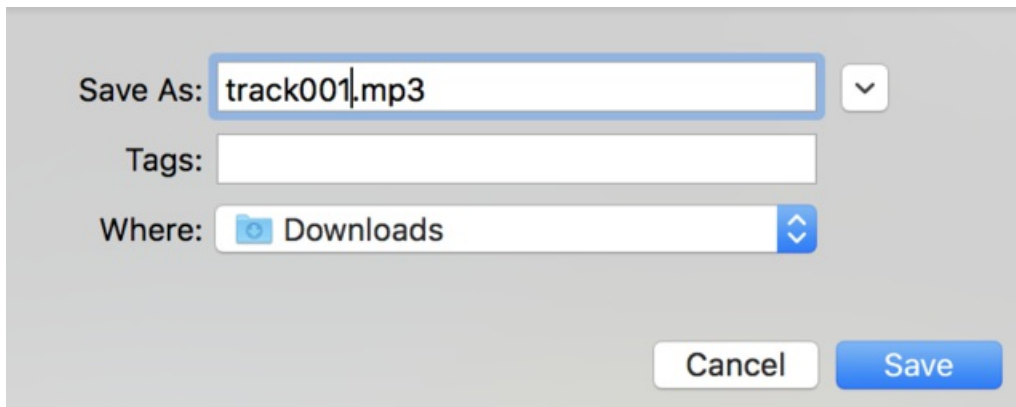
We are going to start by just uploading a single mp3 file. If you don't already have one, don't worry! BartleBeats created a custom AdaBox 004 soundtrack for you to enjoy on your new MusicMaker.

Right click on the button below and click 'Download Linked File'

<https://adafru.it/xdm>

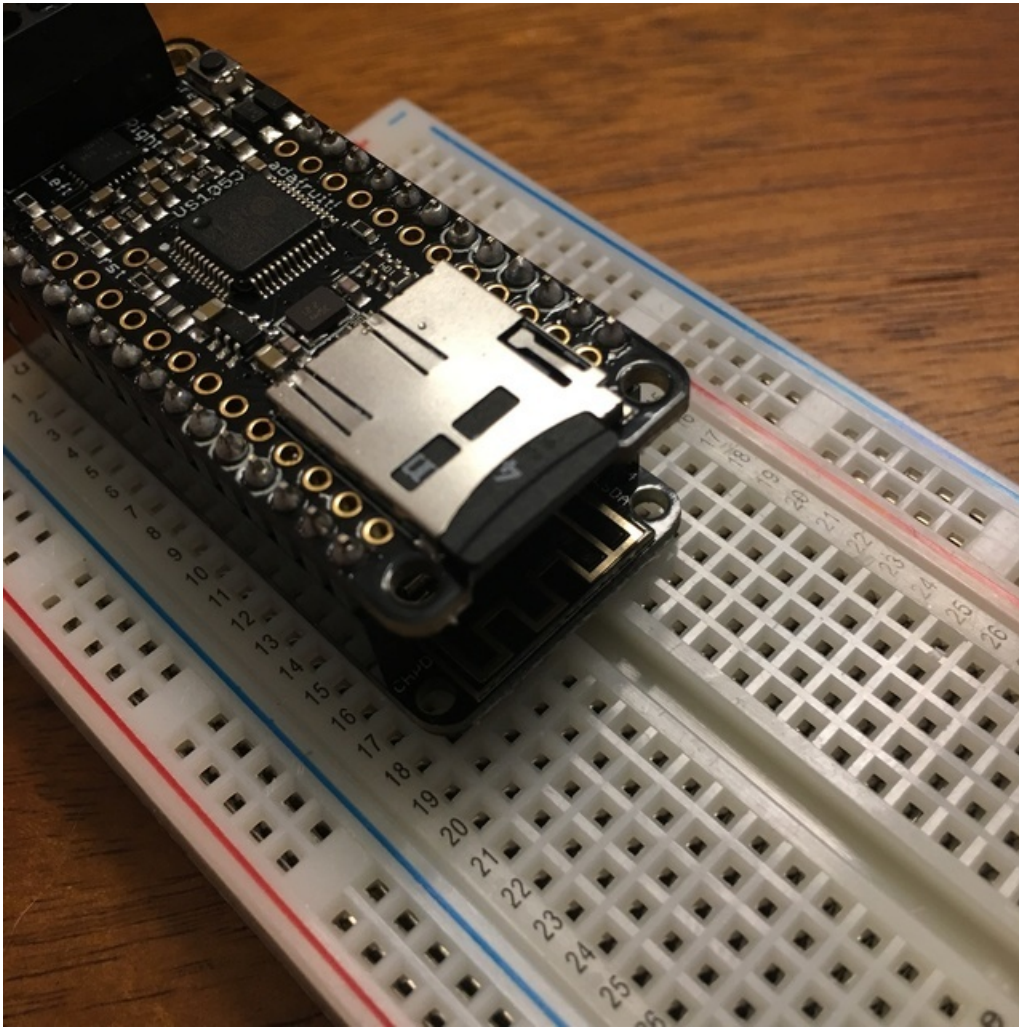
<https://adafru.it/xdm>

This will download an mp3 file called ADABOX004\_-\_Bartlebeats.mp3. In order to get this file to play on the MusicMaker FeatherWing, we need to change the file name to an 8.3 file name. This means 8 letters dot 3 letters. So, lets rename it track001.mp3



Now, copy this track to your SD card. Once it has transferred over, you can install the SD card into the MusicMaker FeatherWing. Push the Micro SD card into the MusicMaker until it clicks into place.





Now we can load some code onto the Huzzah Feather and start playing some music.

## Arduino IDE and Installing New Libraries

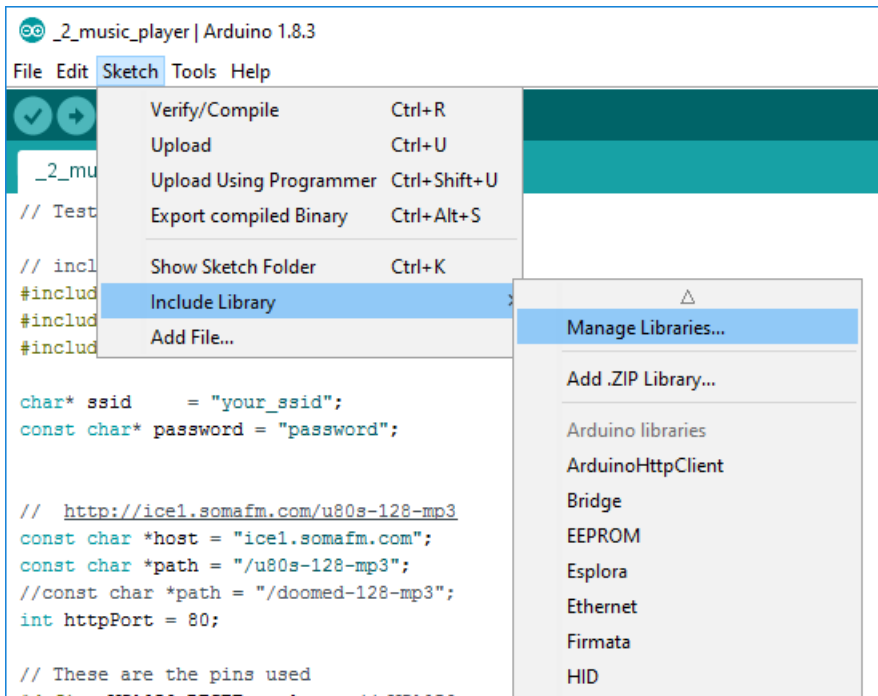
We aren't going to get into too much detail about the Arduino IDE and how to install new libraries, as we have plenty of those on the Adafruit Learning System already.

<https://adafru.it/xdn>

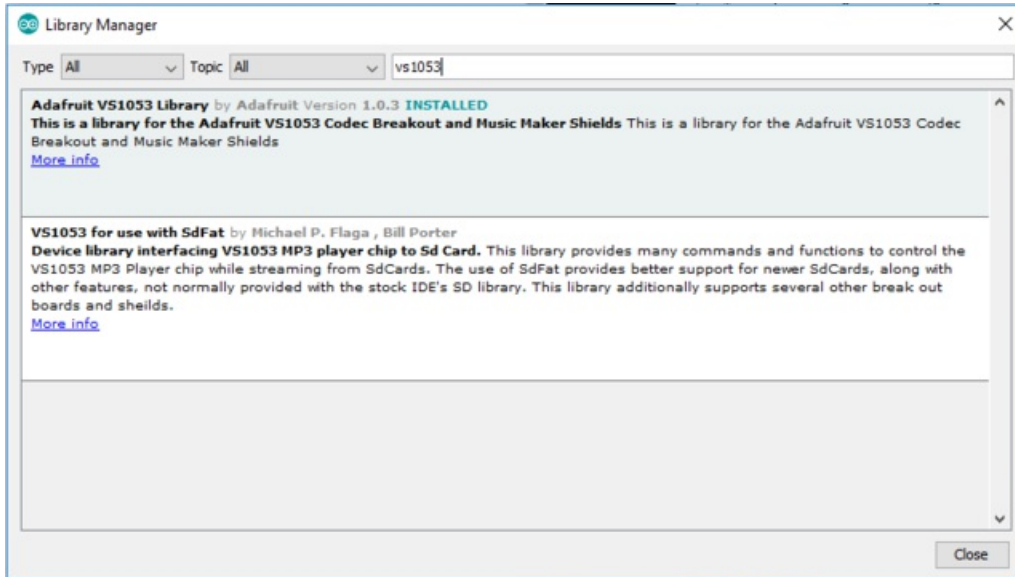
<https://adafru.it/xdn>

The next few pages are going to go over some basic code you can use to get your new HUZAZH Feather and MusicMaker FeatherWing to play nice together. All of them will require some special libraries. You can either use this guide on installing libraries manually, or you can use the latest Arduino IDE Library Manager to do the work for you.

Go to Sketch > Include Library > Manage Libraries...



Then, search for the library you want to install, and click on the correct library name. Then click install. That's it!

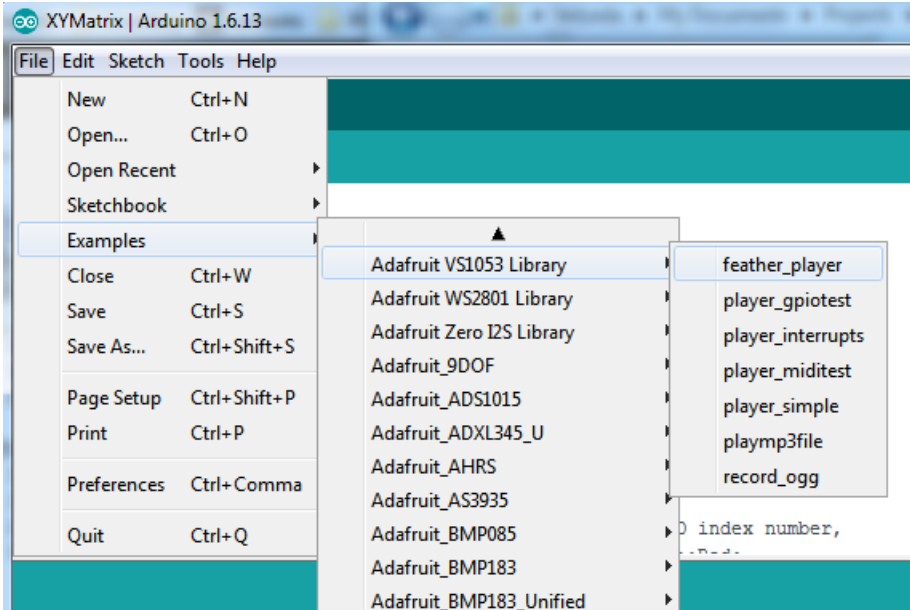


Each of the next pages will tell you which libraries you will need to install. Here we go...

## Playing your first mp3

For the first sketch, we are going to need the Adafruit\_VS1053 library, as well as the SD library. Using the library manager like shown in the previous step, make sure both are installed.

With the VS1053 library installed, we can use the example sketch. Select **File->Examples->Adafruit\_VS1053->feather\_player** to load the example code.



All you need to do is upload that sketch to your Feather and the mp3 should start playing. You can open the serial console to check on the status. It should say something like this (you may need to press the reset button on the MusicMaker FeatherWing with the serial console open to see this):

```
VS1053 found
SD OK!
TRACK001.MP3 14005206
Playing full track 001
```

if you have more files on the SD card, those will also appear. Note the 8.3 naming shows up with the size of the file. Our file is there, and is called **TRACK001.MP3**

## Streaming Internet Radio

This bit of code is going to allow you to stream music over the internet right to your MusicMaker FeatherWing. You will need to have access to WiFi for this to work.

First up, you will need to install two libraries. Check out the previous page for tips on installing libraries. The first library is the Adafruit\_VS1053 library, and the second is the ESP8266WiFi library.

At the top of the code you will see where to enter your WiFi SSID (the name of your WiFi network), and the password. Make sure to leave the quotation marks, and only replace the text between them.

```
char* ssid      = "your_ssid";
const char* password = "password";
```

Below that you will see where you can enter radio stations. For this example, we are using SomaFM to get live, free streaming radio. They're awesome, please donate if you like the streams! The station we are using in the code is an 80's station, but you can just go to [SomaFM.com \(https://adafru.it/xCQ\)](https://adafru.it/xCQ) and pick your own station. When you click on a station, in the left sidebar click on 'Direct Stream Links' and copy the direct server link info for **MP3 128kb**. Make sure to use the same formatting as shown in the code:

```
// http://ice1.somafm.com/u80s-128-mp3
const char *host = "ice1.somafm.com";
const char *path = "/u80s-128-mp3"; //this is the part you need to change
//const char *path = "/doomed-128-mp3";
int httpPort = 80;
```

Of course you can use any mp3-streaming service. Just put in the domain name in **host**, the path to the mp3 stream in **path** and change the **httpPort** if necessary

Here is the full code. Upload this code to your MusicMaker setup and open the serial console. You should see it connect to your network and start playing some music. You will notice that there is a bit of code in there for adding in a potentiometer and on/off switch. We will add these next.

Don't worry if the audio seems to sounds weird - we haven't added an essential potentiometer part so as long as you hear \*something\* keep going!

```
// Tested: ESP8266, ESP32, M0+WINC1500

// include SPI, MP3 and SD libraries
#include <SPI.h>
#include <Adafruit_VS1053.h>
#include <ESP8266WiFi.h>

char* ssid      = "your_ssid";
const char* password = "password";

// http://ice1.somafm.com/u80s-128-mp3
const char *host = "ice1.somafm.com";
const char *path = "/u80s-128-mp3";
//const char *path = "/doomed-128-mp3";
int httpPort = 80;
```

```

// These are the pins used
#define VS1053_RESET -1 // VS1053 reset pin (not used!)
#define VS1053_CS 16 // VS1053 chip select pin (output)
#define VS1053_DCS 15 // VS1053 Data/command select pin (output)
#define VS1053_DREQ 0 // VS1053 Data request, ideally an Interrupt pin

#define VOLUME_KNOB A0
#define ON_OFF_SWITCH 4

int lastvol = 30;

Adafruit_VS1053 musicPlayer = Adafruit_VS1053(VS1053_RESET, VS1053_CS, VS1053_DCS, VS1053_DREQ);

// Use WiFiClient class to create HTTP/TCP connection
WiFiClient client;

void setup() {
  Serial.begin(115200);

  Serial.println("\n\nAdafruit VS1053 Feather WiFi Radio");

  /***** INITIALIZE MP3 WING */
  if (!musicPlayer.begin()) { // initialise the music player
    Serial.println(F("Couldn't find VS1053, do you have the right pins defined?"));
    while (1) delay(10);
  }

  Serial.println(F("VS1053 found"));
  musicPlayer.sineTest(0x44, 500); // Make a tone to indicate VS1053 is working

  // Set volume for left, right channels. lower numbers == louder volume!
  musicPlayer.setVolume(lastvol, lastvol);

  // don't use an IRQ, we'll hand-feed

  pinMode(ON_OFF_SWITCH, INPUT_PULLUP);
  /***** INITIALIZE WIFI */
  Serial.print("Connecting to SSID "); Serial.println(ssid);
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("WiFi connected");
  Serial.println("IP address: "); Serial.println(WiFi.localIP());

  /***** INITIALIZE STREAM */
  Serial.print("connecting to "); Serial.println(host);

  if (!client.connect(host, httpPort)) {
    Serial.println("Connection failed");
    return;
  }

  // We now create a URI for the request
  Serial.print("Requesting URL: ");

```

```

Serial.println(path);

// This will send the request to the server
client.print(String("GET ") + path + " HTTP/1.1\r\n" +
             "Host: " + host + "\r\n" +
             "Connection: close\r\n\r\n");
}

// our little buffer of mp3 data
uint8_t mp3buff[32]; // vs1053 likes 32 bytes at a time

int loopcounter = 0;

void loop() {
  if (! digitalRead(ON_OFF_SWITCH)) {
    yield();
    return;
  }

  loopcounter++;

  // wait till mp3 wants more data
  if (musicPlayer.readyForData()) {
    //Serial.print("ready ");

    //wants more data! check we have something available from the stream
    if (client.available() > 0) {
      //Serial.print("set ");
      // yea! read up to 32 bytes
      uint8_t bytesread = client.read(mp3buff, 32);
      // push to mp3
      musicPlayer.playData(mp3buff, bytesread);

      //Serial.println("stream!");
    }
  } else {
    if (loopcounter >= 1000) {
      loopcounter = 0;
      // adjust volume!
      int vol = 0;
      vol = analogRead(VOLUME_KNOB);
      vol /= 10;
      if (abs(vol - lastvol) > 3) {
        Serial.println(vol);
        lastvol = vol;
        musicPlayer.setVolume(lastvol, lastvol);
      }
    }
  }
}
}

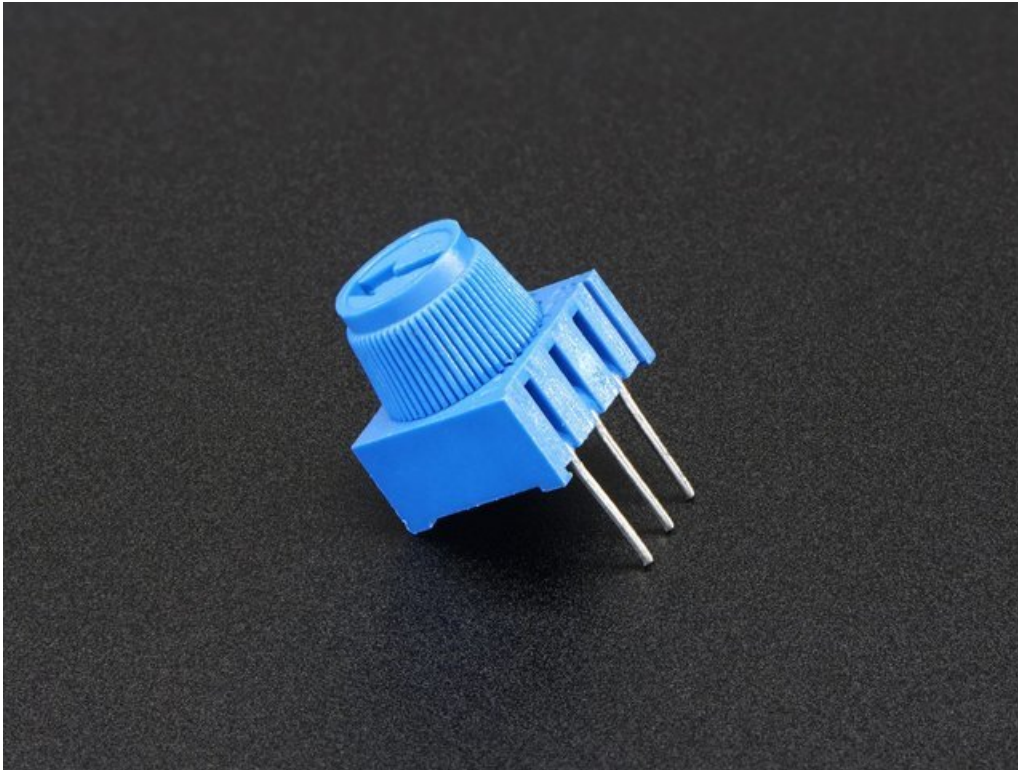
```

## Adding a volume knob

---

In the last step, we got our little MusicMaker setup to stream live internet radio. For this step we will add a potentiometer to the breadboard and use it as a volume knob.

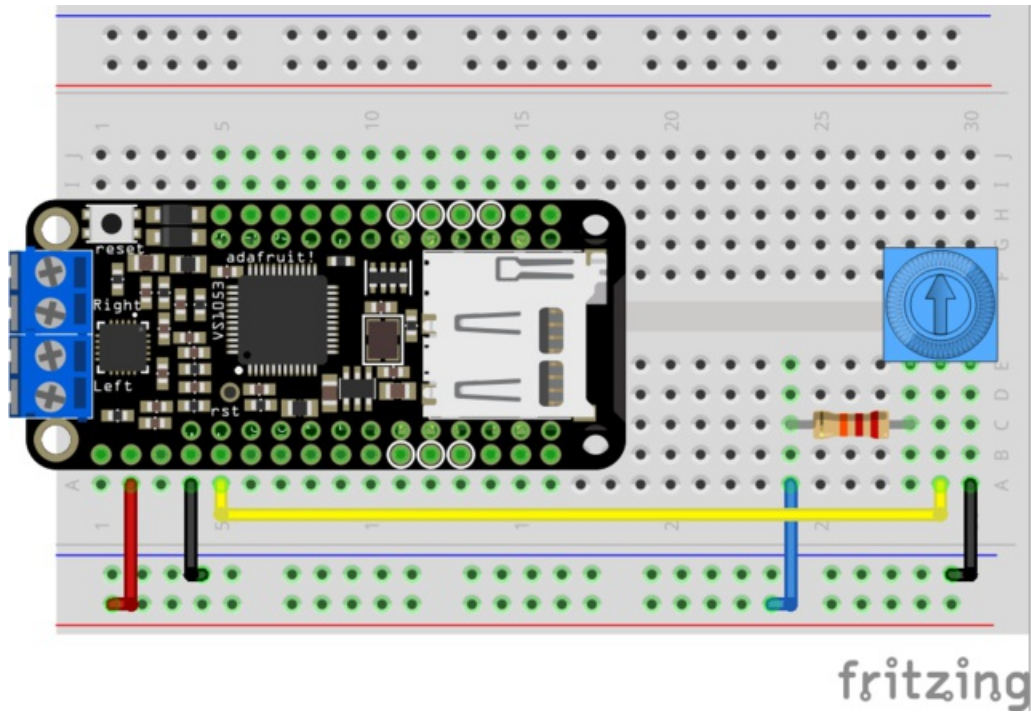
In your AdaBox, grab out the small blue potentiometer:



You will also need to grab the wires and the resistor that was included.

Unplug your radio and wire up your potentiometer and resistor like this:





Now go ahead and power up your little internet radio. Once the music starts playing, twisting the potentiometer should change the volume.

<https://adafru.it/xdU>

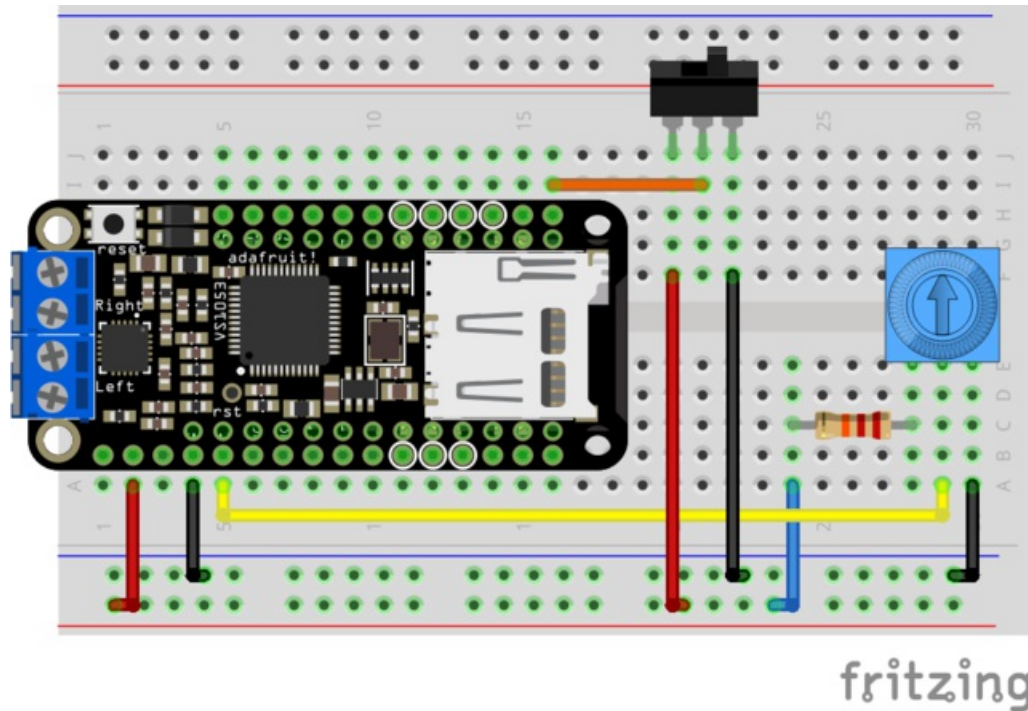
<https://adafru.it/xdU>

Next up, we will add in a power switch so you can turn your streaming radio on and off.

## Adding a power switch

In the last step, we added in a knob to control your internet radio volume. Now, let's add a switch so you can turn your radio on and off.

The code for the power switch is just watching digital pin 4 and if it sees line voltage, it turns on, and if it goes to ground it turns off. Unplug your radio and wire the switch up like this:



Power up your radio and flip the switch to the power side to turn it on. Try flipping the switch once the radio starts playing to see if it shuts off.

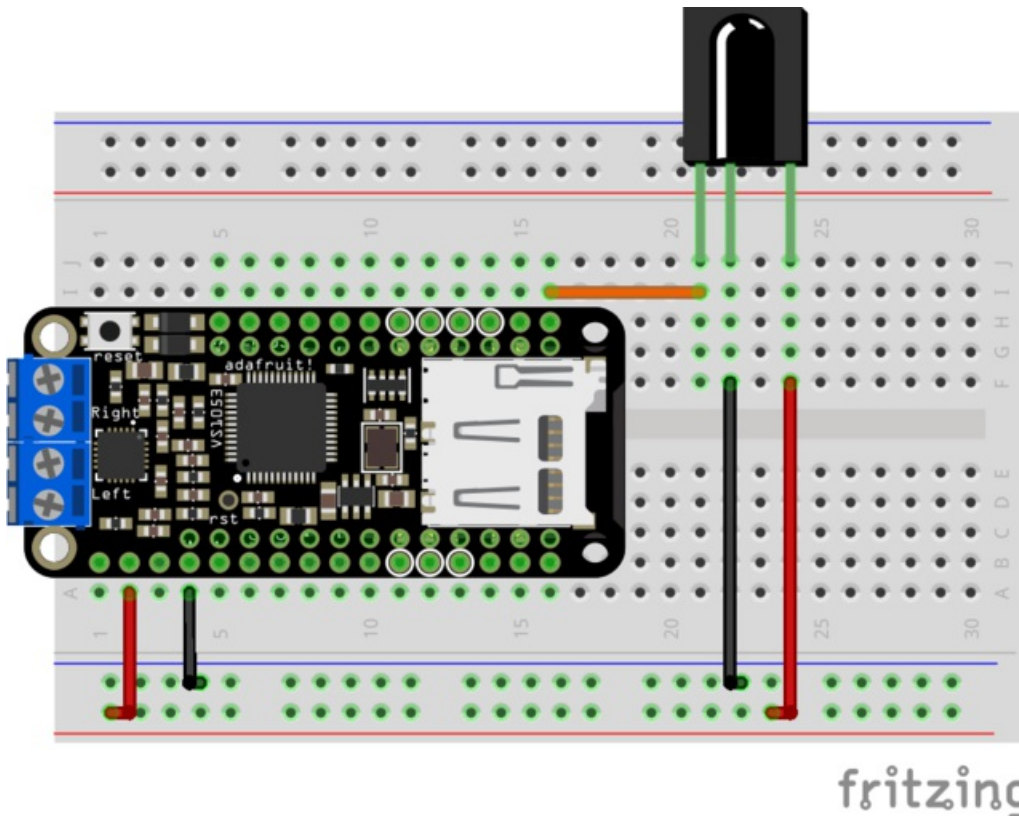
## IR Remote and Receiver



This handy little remote will allow you to control your MusicMaker from a distance. Before going any further, pull the clear plastic piece out of the back of the remote.

In order to read commands from the IR remote, you will need to find the bag labeled IR Sensor. If you did the previous steps, you can clear out the switch, the volume knob, and wires from the breadboard. This remote will handle all of those things. Also, if you want to add more tracks so you can switch between them, the code is currently set up to handle 3 files. Just add more mp3 files with the same naming format (**track001.mp3**, **track002.mp3**, **track003.mp3**, etc).

Install your SD card and wire everything up like this:



For this bit of code you will need to install the **Adafruit\_VS1053** (which you probably already have if you did the other projects) and the IR Remote for ESP8266 library. [You can find the IR library here \(https://adafru.it/xIA\)](https://adafru.it/xIA)

Just upload the code below and open your serial console to watch what happens when you click the buttons on the remote. Press the number 1 on the remote to start that track. Play, pause, and volume keys also work.

```
// include SPI, MP3 and SD libraries
#include <SPI.h>
#include <SD.h>
#include <Adafruit_VS1053.h>
#include <IRrecv.h>
#include <IRutils.h>

int RECV_PIN = 4; //an IR detector/demodulator is connected to GPIO pin 2
IRrecv irrecv(RECV_PIN);

// These are the pins used
#define VS1053_RESET -1 // VS1053 reset pin (not used!)
#define VS1053_CS 16 // VS1053 chip select pin (output)
#define VS1053_DCS 15 // VS1053 Data/command select pin (output)
#define CARDCS 2 // Card chip select pin
#define VS1053_DREQ 0 // VS1053 Data request, ideally an Interrupt pin

Adafruit_VS1053_FilePlayer musicPlayer =
  Adafruit_VS1053_FilePlayer(VS1053_RESET, VS1053_CS, VS1053_DCS, VS1053_DREQ, CARDCS);

// the name of what we're going to play
char foundname[20];
boolean isPaused = false;
uint8_t volume = 10;
int lastRemoteVal = 0;
```

```

int lastRemoteVal = 0;

void setup() {
  Serial.begin(115200);

  Serial.println("\n\nAdafruit VS1053 Feather Test");

  if (!musicPlayer.begin()) { // initialise the music player
    Serial.println(F("Couldn't find VS1053, do you have the right pins defined?"));
    while (1);
  }

  Serial.println(F("VS1053 found"));

  musicPlayer.sineTest(0x44, 500); // Make a tone to indicate VS1053 is working

  if (!SD.begin(CARD_CS)) {
    Serial.println(F("SD failed, or not present"));
    while (1); // don't do anything more
  }
  Serial.println("SD OK!");

  // list files
  printDirectory(SD.open("/"), 0);

  // Set volume for left, right channels. lower numbers == louder volume!
  musicPlayer.setVolume(volume, volume);

  // If DREQ is on an interrupt pin we can do background
  // audio playing
  // musicPlayer.useInterrupt(VS1053_FILEPLAYER_PIN_INT); // DREQ int

  irrecv.enableIRIn(); // Start the receiver
  Serial.println("Playing one Track");
}

void loop() {
  decode_results results;

  if(digitalRead(VS1053_DREQ) && !musicPlayer.stopped() && !isPaused) {
    musicPlayer.feedBuffer();
  }

  // look for a message!
  if (irrecv.decode(&results)) {
    //Serial.println(results.value, HEX);
    irrecv.resume(); // Receive the next value

    // handle repeat codes!
    if (results.value == 0xFFFFFFFF) {
      // only for vol+ or vol-
      if ( (lastRemoteVal == 0xFD40BF) || (lastRemoteVal == 0xFD00FF) )
        results.value = lastRemoteVal;
    } else {
      lastRemoteVal = results.value;
    }

    if (results.value == 0xFD08F7) {
      musicPlayer.stopPlaying();
      Serial.println("playing track #1");
    }
  }
}

```

```

    musicPlayer.startPlayingFile("track001.mp3");
}
if (results.value == 0xFD8877) {
    musicPlayer.stopPlaying();
    Serial.println("playing track #2");
    musicPlayer.startPlayingFile("track002.mp3");
}
if (results.value == 0xFD48B7) {
    musicPlayer.stopPlaying();
    Serial.println("playing track #3");
    musicPlayer.startPlayingFile("track003.mp3");
}

if (results.value == 0xFD40BF) { //vol+
    Serial.println("Vol+");
    if (volume > 0) {
        volume--;
        musicPlayer.setVolume(volume,volume);
    }
}
if (results.value == 0xFD00FF) { //vol-
    Serial.println("Vol-");
    if (volume < 100) {
        volume++;
        musicPlayer.setVolume(volume,volume);
    }
}

if (results.value == 0xFD807F) { // playpause
    Serial.println("Play/Pause");
    isPaused = !isPaused; // toggle!
}
}

delay(1);
}

boolean findFileStartingWith(char *start) {
    File root;
    root = SD.open("/");
    root.rewindDirectory();
    while (true) {
        File entry = root.openNextFile();
        if (! entry) {
            return false;
        }
        String filename = entry.name();
        Serial.print(filename);
        if (entry.isDirectory()) {
            Serial.println("/");
        } else {
            Serial.println();
            if (filename.startsWith(start)) {
                filename.toCharArray(foundname, 20);
                entry.close();
                root.close();
                return true;
            }
        }
    }
}

```

```

    }
    entry.close();
}

/// File listing helper
void printDirectory(File dir, int numTabs) {
    while(true) {

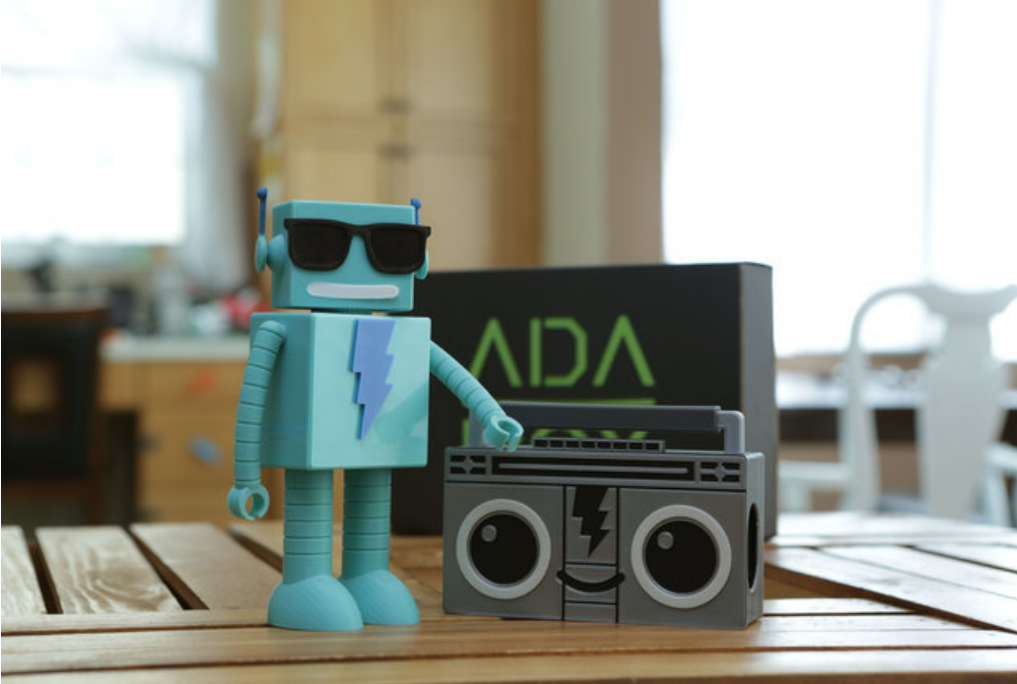
        File entry = dir.openNextFile();
        if (! entry) {
            // no more files
            //Serial.println("***nomorefiles***");
            break;
        }
        for (uint8_t i=0; i<numTabs; i++) {
            Serial.print('\t');
        }
        Serial.print(entry.name());
        if (entry.isDirectory()) {
            Serial.println("/");
            printDirectory(entry, numTabs+1);
        } else {
            // files have sizes, directories do not
            Serial.print("\t\t");
            Serial.println(entry.size(), DEC);
        }
        entry.close();
    }
}
}

```



## AdaBox 004 Projects

---



Now that you have a good understanding of how the individual components of AdaBox 004 work, lets dive in and build some neat projects.

The first project is a way to connect your MusicMaker FeatherWing to Adafruit IO.

The second project is a working 3D printed Boomy the Boombox!

The third project is an AdaBox Upgrade. With a few extra components from the Adafruit Shop (and an old radio) you can build John Park's Haunted Radio project.



## Project 1: WiFi Music Alert Box

---



You can turn the **Internet of Things** into the **Internet of Tunes**! With this easy-to-build project you can use online events to trigger your favorite songs.

The **Feather ESP8266 Huzzah** is a **WiFi** connected board that can use the **Music Maker FeatherWing** with amp to play music at your command. Using the **Adafruit IO** (<https://adafru.it/wDO>) service, a small bit of Arduino code and **IFTTT** (<https://adafru.it/vDp>) (If This then That), events such as **Twitter** mentions, weather reports, **YouTube** uploads, news alerts, and more can trigger songs to play that you have stored on the included SD memory card.

In this great guide from John Park, he will also show you how to take the included box and make it a custom speaker box.

<https://adafru.it/xcn>

<https://adafru.it/xcn>

## Project 2: 3D Printed Boomy the Boombox



Make your very own 3D printed Boomy the Boombox!

For AdaBox subscribers, in each Adabox you'll find super awesome Adafruit Collectible pin. In AdaBox 004, it's Boomy The Boombox. Boomy is a super cool boombox that likes good vibrations and enjoys beat boxing. Boomy loves playing music and likes to reminisce about the 1980's culture.

<https://adafru.it/xco>

<https://adafru.it/xco>

## Project 3: The Haunted Radio



### AdaBox Upgrade!

With just a few extra products from the Adafruit store, you can make your very own Haunted Radio.

Vintage radios are irresistible. When you hear one spewing static, your natural response is to turn the dial and try to tune in a station. The anticipation! What will you hear? With this Haunted Radio project, you can decide for your guests or escape room attendees exactly what they'll hear, and you can even choose the frequency.

There isn't any pirate radio station involved here, either. Instead, the secret is a Feather microcontroller reading the dial position with a magnet sensor, and an attached Music Maker FeatherWing with amplifier to play back your pre-recorded message over the radio's speaker.

<https://adafru.it/xcp>

<https://adafru.it/xcp>

## Bonus Custom BartleBeats AdaBox 004 Soundtrack!

---

As part of the AdaBox004 experience, we asked our in-house musician BartleBeats to create a custom track as a soundtrack for your making! Since this AdaBox is music themed, feel free to share or post your favorite songs that help drive your projects to completion with #adabox And if you'd like to hear more of Adafruit's music, check out [Adafruit's Sound Cloud](https://adafruit.com/music). (<https://adafru.it/xcq>)

<https://adafru.it/xcW>

<https://adafru.it/xcW>

## Share your MusicMaker box designs and get help

We included a blank white box with this AdaBox because we really wanted all of you to create fun and crazy MusicMaker designs. We hope you will share what you have created with us. Below are some of the designs we came up with at Adafruit HQ.

<https://adafru.it/uZD>

<https://adafru.it/uZD>









