



# Flora-Powered TARDIS Costume (for Dogs!)

Created by Anne Barela



<https://learn.adafruit.com/wearable-flora-powered-tardis-costume-dogs>

Last updated on 2024-12-10 02:34:17 PM EST

# Table of Contents

Overview and Parts	3
• Overview	
• Parts List	
Wiring	6
Code	7
The Costume	9
Going Further	11

---

# Overview and Parts

This project is over 11 years old and references Arduino code of the same era. It is suggested that you use this project as an example for your own designs and not rely on every part of this to still work flawlessly as Arduino continues to change their software.



## Overview

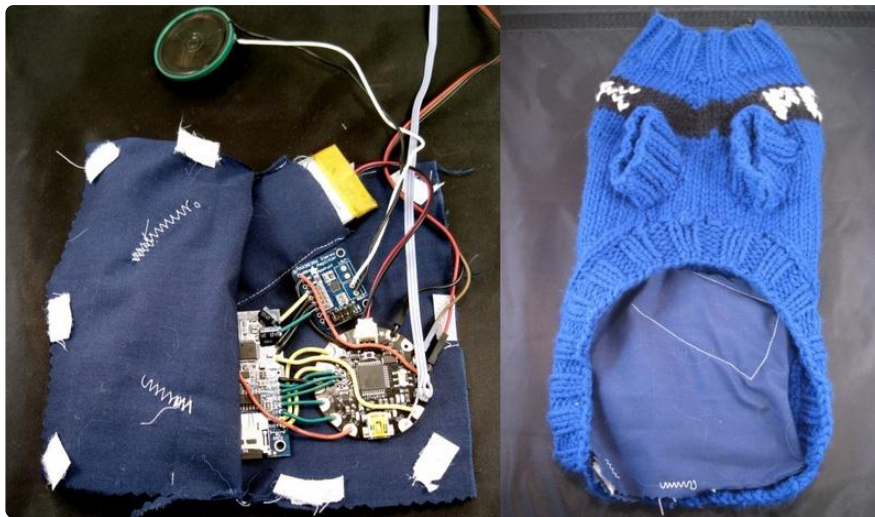
Costuming is a natural for considering adding electronics, especially during Halloween ([Check out the Adafruit Blog's Electric Halloween category for lots of great ideas](https://adafru.it/aRs) (<https://adafru.it/aRs>)).

Everyone knows that Spaniels are adorable, but a Tardis Spaniel is even cuter! Spidey (shown above) is wearing a blue knit Tardis sweater that keeps him warm on October nights and also hides a wearable electronic project that plays the Tardis woop-woop effect and flashes a blue LED as well.

This project uses a [Flora wearable processor](https://adafru.it/cRC) (<https://adafru.it/cRC>) and the [VS1053 MP3 music player module](http://adafru.it/1381) (<http://adafru.it/1381>) which plays the sound of the TARDIS from Doctor Who. A [Flora NeoPixel](http://adafru.it/1260) (<http://adafru.it/1260>) is used for the blinking blue light on top. The project lays flat for a wearable project. [An infrared receiver](http://adafru.it/157) (<http://adafru.it/157>) is added to mute the sound when you want quiet (although you could also use it to switch between different audio tracks)

This project shows the flexibility of flora in making more advanced audio projects. Since a good number of digital pins are required to use the VS1053, this project demonstrates use of the extra pins on the Flora ICSP header for digital signals.





## Parts List

The main components are available in the Adafruit shop. There is flexibility in how you design your project. If you do not need to turn the sound off remotely, you can eliminate the IR sensor and remote. Be careful on adding more Neopixels - powering a small number off the Flora regulator works, if you power a large number you will need to use the battery power or a separate supply and appropriate wiring.

- Adafruit [FLORA](http://adafru.it/659) (<http://adafru.it/659>) wearable microcontroller
- [VS1053 Audio Breakout Board](http://adafru.it/1381) (<http://adafru.it/1381>) (Version 1 or 2) and a [microSD card](http://adafru.it/102) (<http://adafru.it/102>)
- Amplifier Board ([MAX98306](http://adafru.it/987) (<http://adafru.it/987>) shown, the updated [TS2012](http://adafru.it/1552) (<http://adafru.it/1552>) is preferred)
- A [Neopixel](http://adafru.it/1260) (<http://adafru.it/1260>) (alternative would be a blue LED and a resistor nominal 100 ohms)
- [LiPO Battery](http://adafru.it/258) (<http://adafru.it/258>) (choose size based on how long the project needs to run and the weight your project can tolerate). If you do not have one, a [LiPo Charger](http://adafru.it/259) (<http://adafru.it/259>) is required. You can use several batteries and swap them as needed for a longer period.
- Optional [IR Receiver](http://adafru.it/157) (<http://adafru.it/157>) and [Remote Control](http://adafru.it/389) (<http://adafru.it/389>) to turn sound on and off (sound can be irritating for long stretches or if you want to talk). You can use your own remote if you determine the code(s) for the key(s) you use.

You will need the following parts to hook things up:

- [Jumper wire](https://adafru.it/cRx) (<https://adafru.it/cRx>) - stranded will give you flexibility at the expense of a bit more work soldering and stripping ends. You can consider [rainbow wire](https://adafru.it/cRy) (<https://adafru.it/cRy>) for its length and flexibility. For my project, I selected solid as the flex was not crucial and it is a bit easier to work with. A ribbon cable, suggested on [Wearable Wednesday](https://adafru.it/cRz) (<https://adafru.it/cRz>), was

used to connect the Neopixel.

- [Speaker\(s\)](https://adafru.it/caH) (<https://adafru.it/caH>) - the amplifier will produce stereo although mono is fine. For wearables, you want a flat, lightweight speaker although if you have the space, such as a hat, you can use slightly larger speakers which typically sound better at the cost of weight. For a small wearable, the tiny speakers in talking greeting cards (available also at Sparkfun) work well.

## Costume

A knit TARDIS outfit sized for a toy dog was obtained on Etsy at a reasonable price. Vendors have hats, scarves, and other Dr. Who themed items. Sewing your own would also be an easy project as the basic color scheme is blue background, white windows, and black trim.



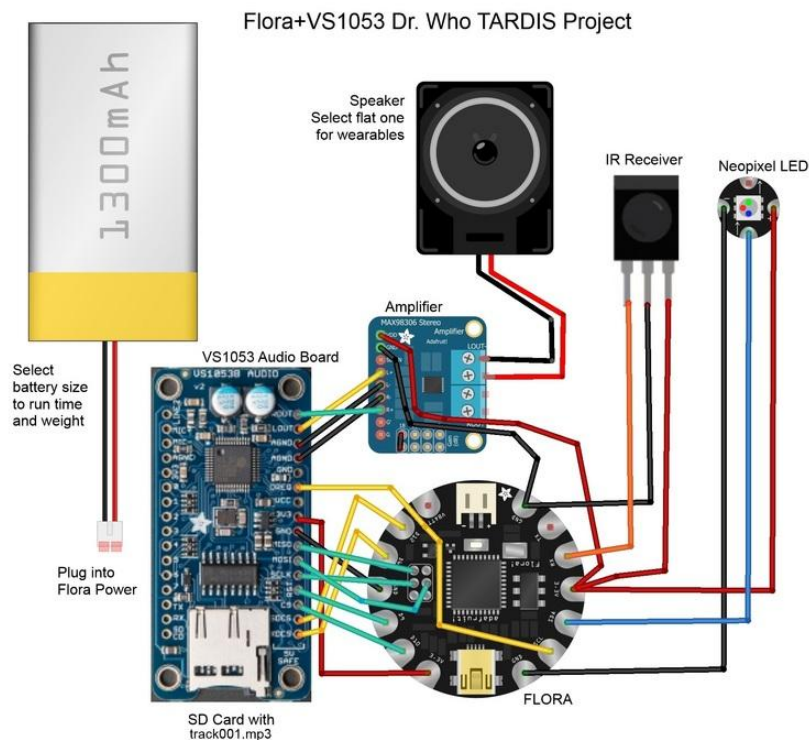
Some color coordinating fabric and thread was used to place the electronics in the costume. Velcro or thread may be used to secure the liner in the costume.

## Sounds

The VS1053 board is very flexible in the sounds it can play (OGG, AAC, WAV, MP3, see the [VS1053 tutorial](https://adafru.it/cRA) (<https://adafru.it/cRA>) for complete details).

The Dr. Who sounds were from [this site](https://adafru.it/19eB) (<https://adafru.it/19eB>) which has a good selection. For your own project, select sound(s) or music that reproduce well on smaller wearable speakers and fit the theme of your project. You can have multiple sounds, even select between them with a bit more coding noted in the Going Further section below. WAV files may be converted to MP3 with the free program Audacity (Windows, Mac, Linux) with the free LAME plug-in. Name your sound track001.mp3 for the code shown.

# Wiring

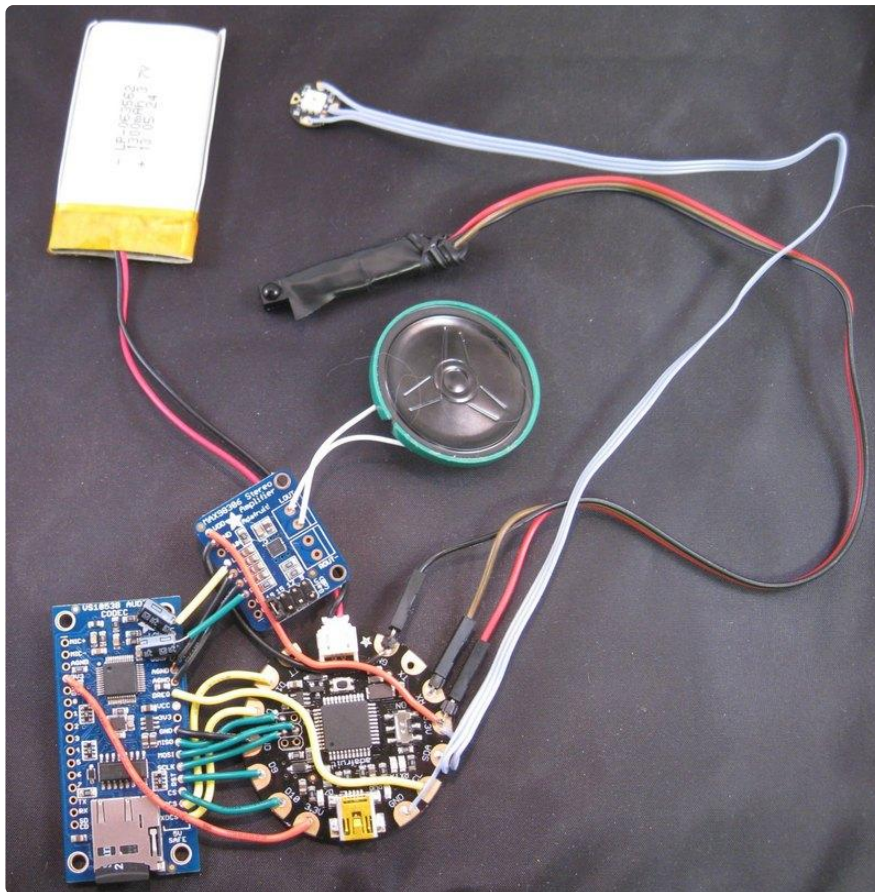


Pin connections and Arduino IDE pin control numbers. Flora follows the Arduino Leonardo pin numbers for SPI: MISO, MOSI, and SCLK which are on Flora's ICSP area (the six holes inside the circle).

Arduino	Flora	VS1053	Other
D0	RX	-	IR Data
D1	TX	-	free
D3	SDA	-	Neopixel
D6	D6	DREQ	
D9	D9	XDCS	
D10	D10	RESET	
D12	D12	CS	
D14	ICSP1/MISO	MISO	
D15	ICSP3/SCLK	SCLK	
D16	ICSP4/MOSI	MOSI	

You can see there is only one digital pin left (TX/D1) if the Neopixel and IR control are used. That is fine for this project, if you wish to add more, the TX pin could be used to time additional Neopixels for example.





Above is the wired circuit, ready to install in the costume.

---

## Code

This project will use the following Arduino libraries that you will want to put into your Arduino Flora library folder:

- [Adafruit VS1053 Library \(https://adafru.it/cR0\)](https://adafru.it/cR0)
- [Adafruit Neopixel Library \(https://adafru.it/aZU\)](https://adafru.it/aZU)
- [Ken Shirriff's IRRemote Library \(https://adafru.it/cR1\)](https://adafru.it/cR1)

It also uses the Arduino IDE built-in SPI bus and SD (flash SD card reader) libraries.

You should consult the Adafruit Guide [All About Arduino Libraries \(https://adafru.it/aYM\)](https://adafru.it/aYM) on how to install the library code for use.

Shirriff's library is known to be well written and it is non-blocking, if there is no IR code received, you can go to do other things. It decodes nearly any remote and is widely used. But it's large: it would never fit on a Trinket or Gemma and takes a fair amount of room on our Flora. But we have the space (with 2100+ bytes to spare), so for the flexibility, it is a good fit.

If you need more space, the serial commands may be commented out. It would make debugging more difficult so consider which statements you need or not.

Use 8 characters + 3 character extension for filenames, re: track001.mp3.

```
/******  
Dr. Who TARDIS Costume Code, based on Adafruit VS1053 code.  
  
Designed to work with the Adafruit VS1053 Codec Breakout  
----&gt; https://www.adafruit.com/products/1381  
  
Adafruit invests time and resources providing this open source code,  
please support Adafruit and open-source hardware by purchasing  
products from Adafruit!  
  
Based on code written by Limor Fried/Ladyada for Adafruit Industries.  
BSD license, all text above must be included in any redistribution  
*****/  
  
// include SPI, MP3 and SD libraries  
#include <SPI.h>;  
#include <Adafruit_VS1053.h>; // Music board library  
#include <SD.h>;  
#include <Adafruit_NeoPixel.h>; // Neopixel Library  
#include <IRremote.h>; // IR library https://github.com/shirriff/Arduino-IRremote  
  
#define PIN 2 // Neopixel on pin D2  
#define NUMPIXELS 1 // One neopixel  
  
Adafruit_NeoPixel strip = Adafruit_NeoPixel(NUMPIXELS, PIN, NEO_GRB + NEO_KHZ800);  
  
// define the pins used  
#define RESET 9 // VS1053 reset pin (output)  
#define CS 10 // VS1053 chip select pin (output)  
#define DCS 6 // VS1053 Data/command select pin (output) (XDCS)  
#define DREQ 3 // VS1053 Data request pin (into Arduino)  
#define CARDCS 12 // Card chip select pin (SDCS)  
  
Adafruit_VS1053_FilePlayer musicPlayer = Adafruit_VS1053_FilePlayer(RESET, CS, DCS, DREQ, CARDCS);  
  
int RECV_PIN = 0; // IR Receiver data line on pin D0  
IRrecv irrecv(RECV_PIN); // initialize IR Remote library  
decode_results results; // decode results variable  
int8_t sound_status; // variable to determine whether to play sound or not  
  
void setup() { // Initialize the hardware  
  Serial.begin(9600);  
  // initialise the music player  
  if (!musicPlayer.begin()) {  
    Serial.println("VS1053 music board not found");  
    while (1); // don't do anything more  
  }  
  
  musicPlayer.sineTest(0x44, 500); // Let person know it's working  
  
  if (!SD.begin(CARDCS)) {  
    Serial.println("SD failed, or not present");  
  }  
}
```



```

    while (1); // don't do anything more
}

musicPlayer.setVolume(20,20);
musicPlayer.useInterrupt(VS1053_FILEPLAYER_PIN_INT);

strip.begin(); // Initialize Neopixel library use
strip.show(); // Initialize all pixels to 'off'

irrecv.enableIRIn(); // Start the IR receiver
sound_status = 1; // we want sound
}

void loop() {

    // Start playing a file, flash LED and listen to IR while playing
    if( sound_status == 1 ) { // if we want sound on
        if (! musicPlayer.startPlayingFile("track001.mp3")) {
            Serial.print("Could not open file");
        }
        Serial.println("Started playing");
    }

    while (musicPlayer.playingMusic) {
        // file is now playing in the 'background' so now's a good time
        // to pulse the LED and look for the IR signal
        sound_and_IR();
    }
    if( sound_status == 0 ) { // work to do even if sound is not playing
        sound_and_IR();
    }
    Serial.print("Done playing music, sound status = ");
    Serial.println(sound_status, DEC);
}

void sound_and_IR(void) { // ths is the code that is run during a sound or if
sound is turned off
    int i;
    for(i=0; i<256; i++) {
        strip.setPixelColor(0,strip.Color(0,0,i)); // cycle Blue up
        strip.show();
        delay(6);
    }
    delay(100);
    for(i=255; i>=0; i--) {
        strip.setPixelColor(0,strip.Color(0,0,i)); // cycle Blue down
        strip.show();
        delay(5);
    }
    delay(100);
    if (irrecv.decode(&results)) { // If any code received by remote
        if( sound_status == 1 ) // if sound status is on, toggle off
            sound_status=0;
        else // if sound status if off, toggle on
            sound_status=1;
        Serial.println(results.value, HEX); // Print out remote hex code
        irrecv.resume(); // Reset to receive another IR value
    }
}
}

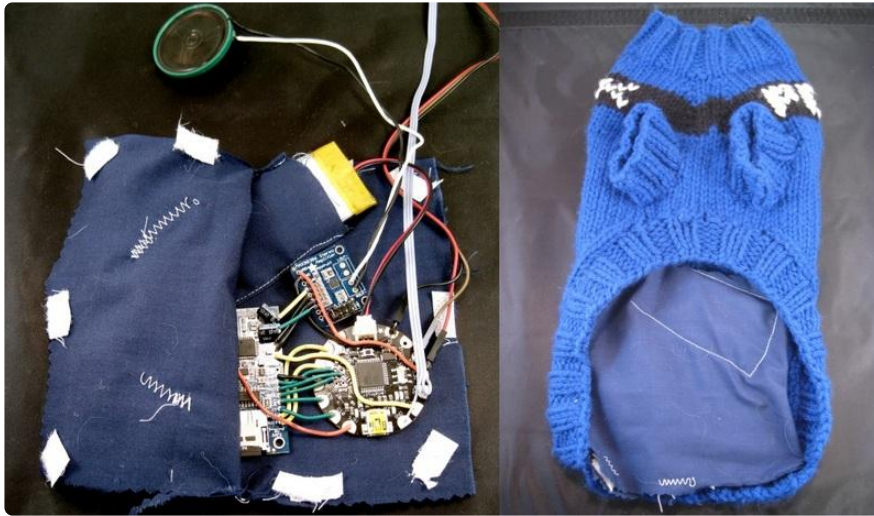
```

---

## The Costume

To keep the electronics secure in the knit outfit, a 5 inch square bag (5" x 10" of cloth) is fashioned. Velcro was used to seal up the top and bottom. The LiPo is the heaviest

part - to support the parts and protect the wearer, a small inner bag was sewn in. The Flora on/off switch is accessible from the opening at the bottom of the picture while the speaker, IR sensor, and Neopixel are fed out the opposite end into the outfit. The loopy side of the velcro is sewn to hold the bag to the knit outfit with a zigzag stitch.



Here is the completed project and a close-up of the Neopixel at the top and the IR sensor in the center of the black band (blending in).



And here is an English Toy Spaniel ("Spidey") wearing the costume:



And video with the sound effects (Spidey was fairly tolerant of the sound after becoming a bit familiar with it but it did help to be able to turn it off after a bit):

---

## Going Further

The basic circuitry can be reused in a number of costume ideas:

- For the December holidays, a seasonal tune may accompany a blue light for Hanukkah or a twinkling star for the top of a Christmas tree.
- The electronics fits easily in a moderate or larger size hat.

As far as music and sound, this project just loops one MP3. You can have additional music tracks (track002.mp3, track003.mp3 and so forth) and play them either sequentially, or my favorite: trigger different sound tracks based on different buttons pressed on the remote control.

To use your own remote and differentiate buttons:

1. Ensure the serial commands are printing to the Arduino IDE serial console.
2. When a sound is playing and you press a button, the hexadecimal number for the key press will be displayed.
3. Write that down. Find the others you want the same way.
4. Generate code similar to that below (assuming the two hex values represent the codes you wrote down earlier):

```
char track_name[13];  
...  
if(results.value == 0xDEADBEEF) { // code was found to be the "1" key  
  track_name="track001.mp3";  
} else if(results.value == 0xFEED1234) { // code was found to be the "2" key  
  track_name="track002.mp3";  
}
```



Your key values will vary. These codes are examples only and do not match any particular remote.

Then to play the correct file, replace

```
musicPlayer.startPlayingFile("track001.mp3")  
    with  
musicPlayer.startPlayingFile(track_name)
```

If you want to simplify the project, you can remove the NeoPixel and code and/or remove the IR receiver and code. If you do not have a way to mute the sound, it could be annoying after a while listening to the same tune.