



# Improving our ESP32 MP3 Player

🕒 June 17, 2020    👤 XTronical    📁 Audio    💬 1

We saw in the last instalment ([here](#)) how to simply play an MP3 on our ESP32. In this article/video we look at how to add volume control and to play all the MP3's that we put on the root directory of our SD card (ignoring anything else). The video below takes you through all this, in addition the source code and affiliate links are also available further down.

**Affiliate links** for the items shown: Clicking these costs you the same price but gives me a small commission, please consider supporting this channel by using them 😊

- MAX98357A : <https://amzn.to/3eovdrd>
- (You will need two of these for stereo sound)
- Potentiometer : <https://amzn.to/3dDBorD>
- ESP32 : <https://amzn.to/2Xzhc3k>
- Breadboards: <https://amzn.to/2THZTVy>
- Speakers : <https://amzn.to/2zBhK0F>
- 16Gb SanDisk SD card : <https://amzn.to/2XH1sLA>
- SD Card Reader (<https://amzn.to/2ApOppK>), note this is the only one I could quickly find that said it supported 3.3v and it's for the normal size SD cards not Micro, you could always by a micro one and do my hack

**The Demo Code** (Full code with volume and ability to cycle through all MP3's on root of SD card).

```
1 // will play MP3's from the root of an SD card, ignoring other files
2 // By XTronical, www.xtronical.com, use as you wish
3 // Based on work and on the audio library of schreibfaul1
4 // See github page : https://github.com/schreibfaul1/ESP32-audioI2S
5 // Also has volume control via a potentiometer attached to pin 13
6
7 #include "Arduino.h"
8 #include "Audio.h"
9 #include "SD.h"
10 #include "FS.h"
11
12 // Digital I/O used
13 #define SD_CS      5
14 #define SPI_MOSI   23 // SD Card
15 #define SPI_MISO   19
16 #define SPI_SCK    18
17
18 #define I2S_DOUT    25
19 #define I2S_BCLK    27 // I2S
20 #define I2S_LRC     26
21
22 #define VolPin      13
23
24 Audio audio;
25
26 uint8_t Volume; // range is 0 to 21
```

We use cookies to ensure that we give you the best experience on our website. If you continue to use this site we will assume that you are happy with it.

Ok

SEARCH ...

SUPPORT THIS SITE

## Support XTronical and its content

\$1.00

Give any amount you wish to say thank you or help support this site.

Donate

SPONSORED

```

32     digitalWrite(SD_CS, HIGH);
33     SPI.begin(SPI_SCK, SPI_MISO, SPI_MOSI);
34     Serial.begin(115200);
35     if(!SD.begin(SD_CS))
36     {
37         Serial.println("Error talking to SD card!");
38         while(true); // end program
39     }
40     audio.setPinout(I2S_BCLK, I2S_LRC, I2S_DOUT);
41     RootDir = SD.open("/");
42     PlayNextSong(); // Play next song, which will be the first at
43 }
44
45 void loop()
46 {
47     audio.loop();
48     audio.setVolume(GetVolume()); // Check volume level and adjust if necessary
49 }
50
51
52 void audio_eof_mp3(const char *info){ //end of file
53     PlayNextSong();
54 }
55
56 void PlayNextSong()
57 {
58     bool SongFound=false;
59     bool DirRewound=false;
60
61     while(SongFound==false)
62     {
63         File entry = RootDir.openNextFile();
64         if (!entry) // no more files
65         {
66             if(DirRewound==true) // If we've already rewound once then there are r
67             {
68                 Serial.println("No MP3 files found to play");
69                 entry.close();
70                 return;
71             }
72             //else we've reached the end of all files in this directory, just rewind back to be
73             RootDir.rewindDirectory(); // reset back to beginning
74             DirRewound=true; // Flag that we've rewound
75         }
76         else
77         {
78             if (!entry.isDirectory()) // only enter this if not a DIR
79             {
80                 if(MusicFile(entry.name())) // Only enter if one of the accepte
81                 {
82                     Serial.print("Playing ");Serial.println(entry.name());
83                     audio.connecttoSD(entry.name()); // Play the file
84                     SongFound=true;
85                 }
86             }
87         }
88         entry.close();
89     }
90 }
91
92 bool MusicFile(String FileName)
93 {
94     // returns true if file is one of the supported file types, i.e. mp3,aac
95     String ext;
96     ext=FileName.substring(FileName.indexOf('.')+1);
97     if((ext=="mp3")||(ext=="aac"))
98         return true;
99     else
100         return false;
101 }
102
103
104
105 uint8_t GetVolume()
106 {
107     // looks at the ADC pin that the potentiometer is connected to.
108     // returns the value as a volume setting
109     // The esp32's ADC has linearity problems at top and bottom we will ignore them and or
110
111     uint16_t VolumeSettingReading;
112
113     VolumeSettingReading=analogRead(VolPin);
114     if(VolumeSettingReading<25) // because of problems mentioned above, anything below 25
115         return 0;
116     if(VolumeSettingReading>4000) // because of problems mentioned above, anything above 40
117         return 21;
118     // If we get this far we are in the middle range that should be linear 500-4000
119     return uint8_t(((VolumeSettingReading-25)/190)); // this will give the correct 0-21 rd
120 }

```

 AUDIO I2S

« **PREVIOUS**

ILI9341 SD card or other problems with ESP32? Try this.

**NEXT** »

ESP32 – Intro to I2S Part 1