

```

/*
 * Call for swallows and swifts
 * Arduino Code by Fabrizio Calderan, 2022.03.20
 *
 * Requirements:
 *
 * - 1x ESP8266/12E
 * - 1x DFPlayer Mini
 * - 1x 3W/4Ω Speaker
 * - 1x On-Off switch
 * - 1x 10KΩ Trimmer
 * - 1x 120x80x50 IP56 enclosure box for outdoor usage.
 */

#include "SoftwareSerial.h"
#include "DFRobotDFPlayerMini.h"

// MP3 Serial communication
SoftwareSerial mySoftwareSerial(14, 12); //12, 13?
DFRobotDFPlayerMini myDFPlayer;

// Variables
int volume;
bool IDLE = true;

// Playback state get information through the BUSY pin
bool playbackState = digitalRead(13);

void setup() {

    mySoftwareSerial.begin(9600);
    // Start serial connection
    Serial.begin(115200);
    delay(500);
    Serial.println("Init MP3 module");
    // Use softwareSerial to begin communication with the MP3 module
    if (!myDFPlayer.begin(mySoftwareSerial, false)) {
        while(true) {
            delay(0); // ESP8266 watchdog needs this
        }
    }

    /*
     * Set the volume
     */
    setVolume();
    Serial.println("----- /setup -----");
}

void loop() {

    setVolume();

```

```

    playbackState = digitalRead(13);

    /*
     * LOW = The MP3 module is busy
     * HIGH = The MP3 module is available
     */

    if (playbackState == HIGH) {
        IDLE = true;
        myDFPlayer.stop();
        myDFPlayer.play(1);
    }
    else {
        if (IDLE == false) {
            myDFPlayer.pause();
            myDFPlayer.stop();
            Serial.println("End of playback");
        }

        IDLE = true;
        delay(200);
    }

    Serial.println("----- /loop -----");
}

```

```

void setVolume() {

    // Get the value of potentiometer in the range of [-1..30]
    volume = map(analogRead(A0), 0, 1023, -1, 30);
    volume = constrain(volume, -1, 30);
    if (volume < 0) {
        volume = 25;
    }

    // Set the player volume
    myDFPlayer.volume(volume);
    Serial.print("Volume level: ");
    Serial.println(String(volume));
}

```