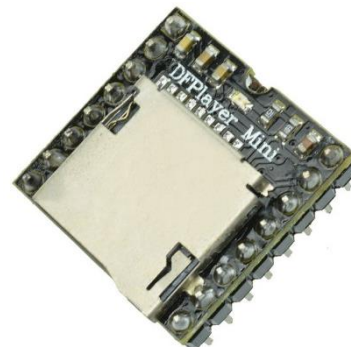


## Modul s integrovaným přehrávačem audia

### 1. POPIS

Modul je osazen přehrávačem audio souborů, který je schopen přehrát standardní audio formáty. Součástí zařízení je také microSD slot, který slouží pro čtení audio souborů z TF úložiště. Přehrávač lze také osadit USB portem pro připojení flash disku. Modul je schopen pracovat se souborovými systémy FAT 16 a 32. Zařízení dále podporuje komunikační rozhraní UART, obsahuje výstup pro sluchátka nebo zesilovač. Modul lze osadit i reproduktory do výkonu 3 W bez použití zesilovače. Dále je možné přístroj osadit tlačítky pro ovládání.



Základní charakteristika:

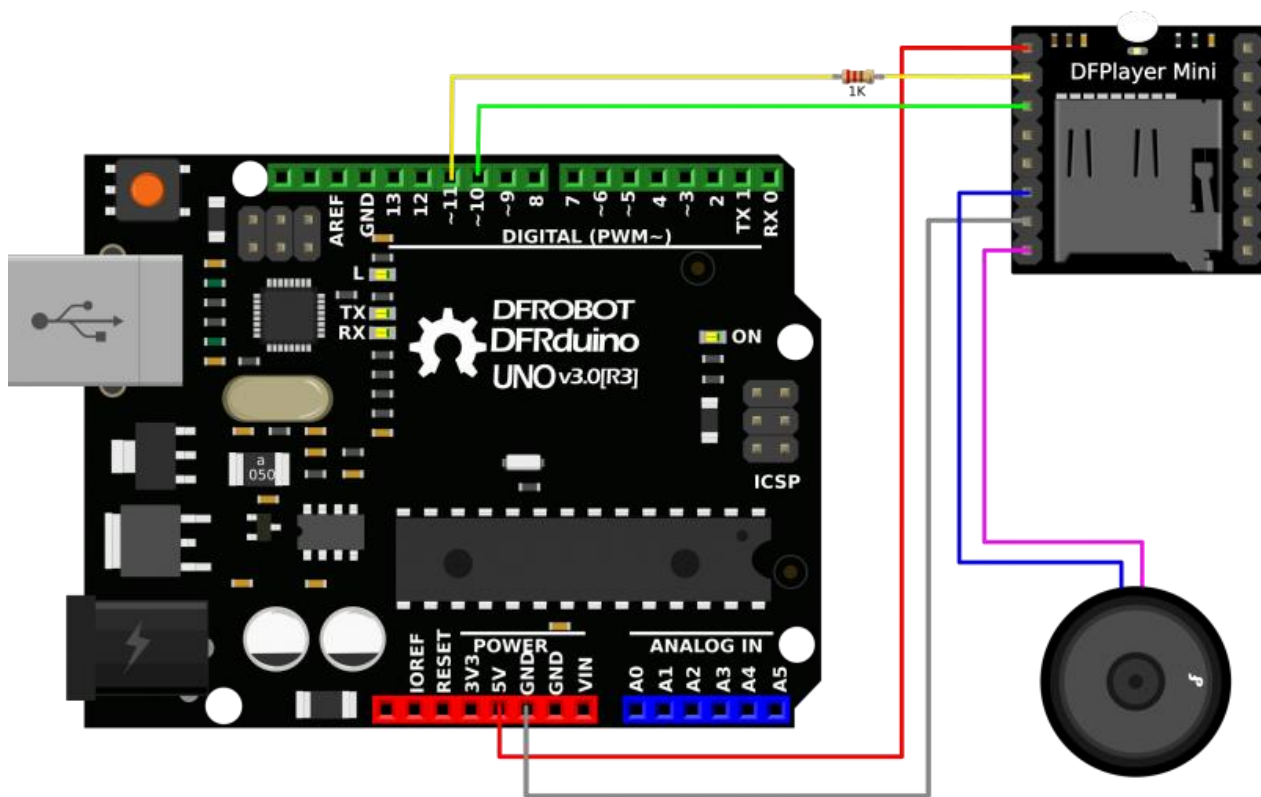
- přehrává formáty MP3, WAV a WMA
- podporovaný souborový systém FAT 16 a 32 (až 32 GB)
- microSD slot, možnost rozšíření o USB port
- vstupy pro tlačítka
- výstup pro reproduktory do 3 W, sluchátka a zesilovač
- UART rozhraní

### 2. SPECIFIKACE

<b>Napájení</b>	3,2 až 5 V	<b>Komunikační rychlost</b>	9600 bps
<b>DA převodník</b>	24 bitů	<b>Počet souborů ve složce</b>	255
<b>Souborové systémy</b>	FAT 16 a 32	<b>Počet složek</b>	100
<b>Max. velikost úložiště</b>	32 GB	<b>Dynamický rozsah</b>	90 dB
<b>MicroSD slot</b>	Ano	<b>SNR</b>	85 dB
<b>Příprava pro USB port</b>	Ano	<b>Vzorkovací frekvence</b>	od 8 do 48 kHz
<b>UART</b>	Ano	<b>Rozměry (mm)</b>	21 x 21



### 3. ZAPOJENÍ



Číslo pinu	Název pinu	Funkce
1	VCC	Napájení (+) 3,2 až 5 V
2	RX	UART přijímací pin
3	TX	UART vysílací pin
4	DAC_R	Pravý výstup pro sluchátka a zesilovač
5	DAC_L	Levý výstup pro sluchátka a zesilovač
6	SPK1	Výstup pro reproduktor do 3 W
7	GND (2X)	Napájení (-), zem
8	SPK2	Výstup pro reproduktor do 3 W

Číslo pinu	Název pinu	Funkce
9	IO1	Tlačítko – přechází skladba, snížení hlasitosti
10	GND (2X)	Napájení (-), zem
11	IO2	Tlačítko – další skladba, zvýšení hlasitosti
12	ADKEY1	Spoušť – první segment
13	ADKEY2	Spoušť – pátý segment
14	USB+	USB port +
15	USB-	USB port -
16	BUSY	Status zařízení

## 00101 01001 00001 4. UKÁZKA PROGRAMU

Kód byl převzat z wiki stránky výrobce modulu. Pro správnou funkci je nutné stáhnout knihovnu [DFRobotDFPlayerMini.h](#).

```
#include "Arduino.h"
#include "SoftwareSerial.h"
#include "DFRobotDFPlayerMini.h"

SoftwareSerial mySoftwareSerial(10, 11); // RX, TX
DFRobotDFPlayerMini myDFPlayer;
void printDetail(uint8_t type, int value);

void setup()
{
  mySoftwareSerial.begin(9600);
  Serial.begin(115200);

  Serial.println();
  Serial.println(F("DFRobot DFPlayer Mini Demo"));
  Serial.println(F("Initializing DFPlayer ... (May take 3~5 seconds)"));

  if (!myDFPlayer.begin(mySoftwareSerial)) { //Use softwareSerial to communicate with mp3.
    Serial.println(F("Unable to begin:"));
    Serial.println(F("1.Please recheck the connection!"));
    Serial.println(F("2.Please insert the SD card!"));
    while(true);
  }
  Serial.println(F("DFPlayer Mini online."));

  myDFPlayer.volume(10); //Set volume value. From 0 to 30
  myDFPlayer.play(1); //Play the first mp3
}
```

```

void loop()
{
    static unsigned long timer = millis();

    if (millis() - timer > 3000) {
        timer = millis();
        myDFPlayer.next(); //Play next mp3 every 3 second.
    }

    if (myDFPlayer.available()) {
        printDetail(myDFPlayer.readType(), myDFPlayer.read());
    }
}

void printDetail(uint8_t type, int value){
    switch (type) {
        case Timeout:
            Serial.println(F("Time Out!"));
            break;
        case WrongStack:
            Serial.println(F("Stack Wrong!"));
            break;
        case DFPlayerCardInserted:
            Serial.println(F("Card Inserted!"));
            break;
        case DFPlayerCardRemoved:
            Serial.println(F("Card Removed!"));
            break;
        case DFPlayerCardOnline:
            Serial.println(F("Card Online!"));
            break;
        case DFPlayerPlayFinished:
            Serial.print(F("Number:"));
            Serial.print(value);
            Serial.println(F(" Play Finished!"));
            break;
        case DFPlayerError:
            Serial.print(F("DFPlayerError:"));
            switch (value) {
                case Busy:
                    Serial.println(F("Card not found"));
                    break;
                case Sleeping:
                    Serial.println(F("Sleeping"));
                    break;
                case SerialWrongStack:
                    Serial.println(F("Get Wrong Stack"));
                    break;
                case CheckSumNotMatch:
                    Serial.println(F("Check Sum Not Match"));
                    break;
                case FileIndexOut:
                    Serial.println(F("File Index Out of Bound"));
                    break;
                case FileMismatch:
                    Serial.println(F("Cannot Find File"));
                    break;
                case Advertise:
                    Serial.println(F("In Advertise"));
                    break;
                default:
                    break;
            }
            break;
        default:
            break;
    }
}

```