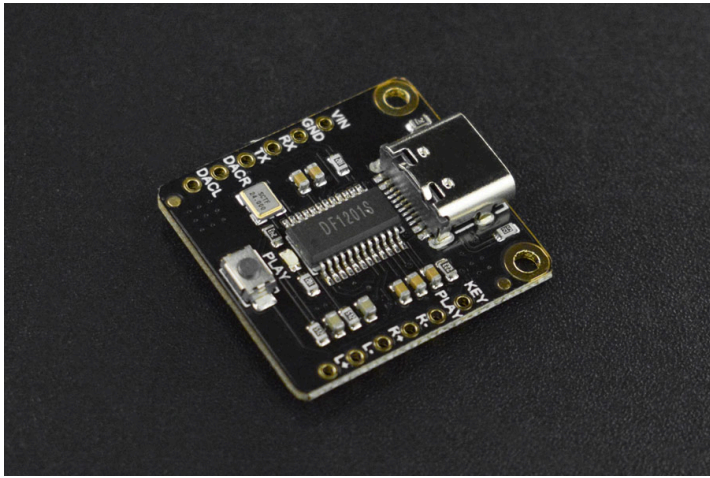


SKU:DFR0768 (<https://www.dfrobot.com/product-2232.html>)



([https://www.dfrobot.com/product-](https://www.dfrobot.com/product-2232.html)

[2232.html](https://www.dfrobot.com/product-2232.html))

Introduction

Here comes the DFPlayer Pro—a mini simple but powerful MP3 Player! This MP3 player module supports four controlling modes: Arduino, AT command, on-board buttons, and ADKEY (<https://www.dfrobot.com/product-2267.html>). You can directly press the on-board button to play or switch music without using a controller. By using a USB cable, you can easily copy your favorite songs into this module to play them any where you want, or use it as a sound card for your PC or Raspberry Pi after connecting them together.

Compared with the previous DFPlayer Mini, the DFPlayer Pro has 128MB storage that can be directly used without a TF card, and the on-board Type-C socket makes it much convenient to use. And it offers dual channel output, better sound effects. Besides that, DFPlayer Pro is equipped with new functions like fast-forward, fast-rewind, play the audio from a particular point of the audio file, etc.

Features

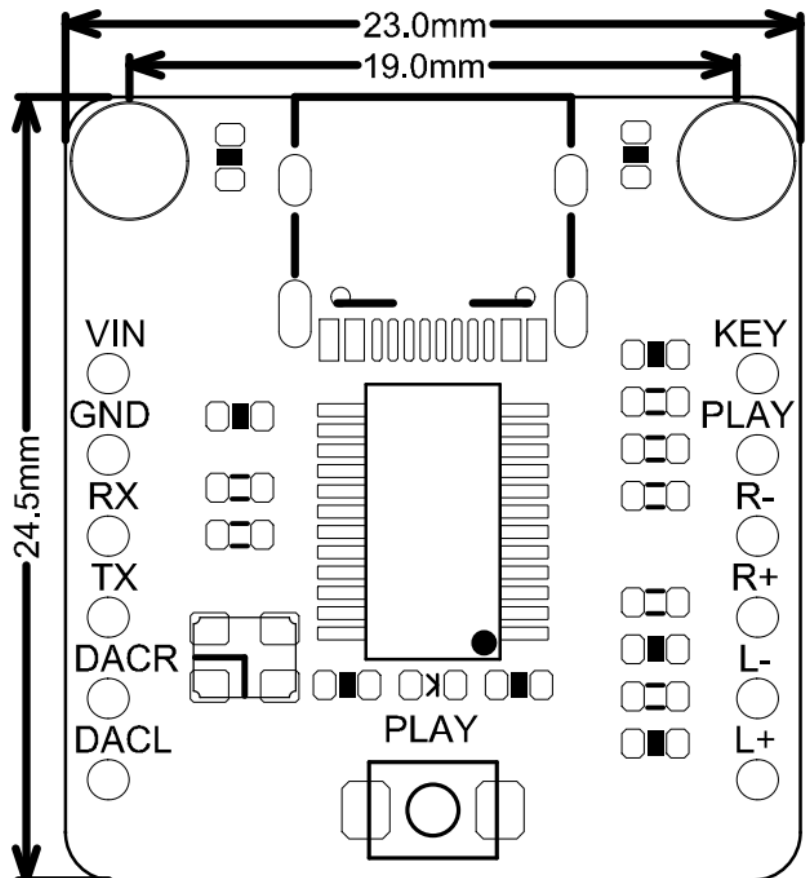
- On-board 128MB High-speed Storage
- Playback Switch by Button
- Dual Channel 3W Power Amplifier
- Support analog U-disk

Application

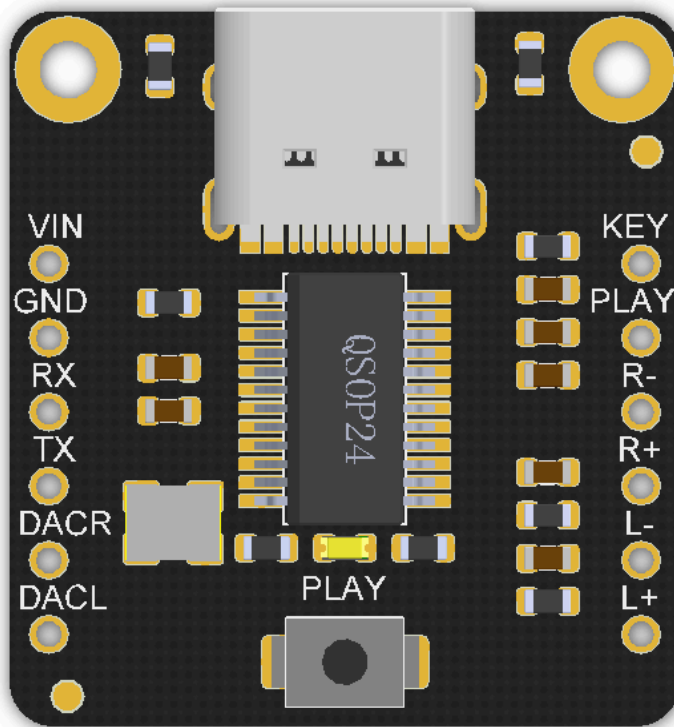
- DIY Plush Toys
- Handmade Personalized Gift
- Musical Alarm Clock
- USB Sound Card

Specification

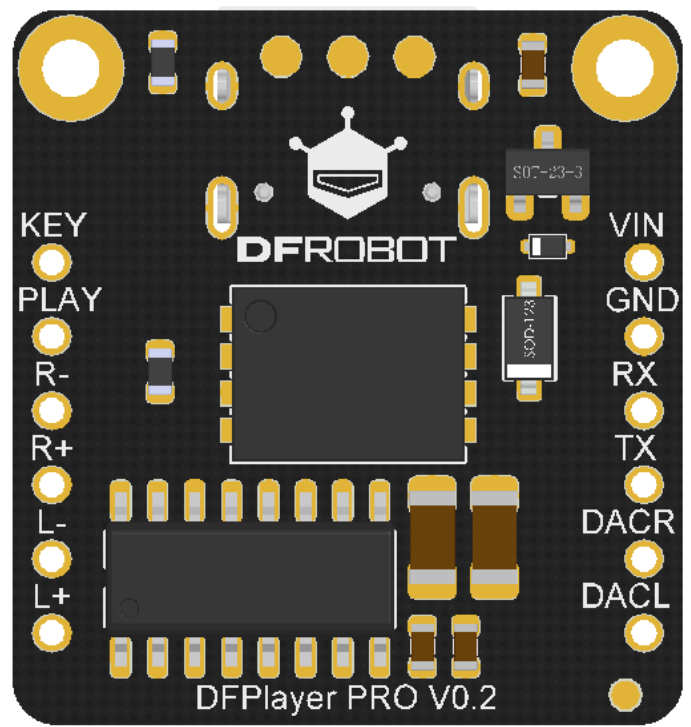
- Power Supply: 3.3V~5V
- Operating Current: >20mA
- Storage: 128MB
- Audio Format: MP3, WAV, WMA, FLAC, AAC, APE
- Communication: UART
- Operating Temperature: -20°C~70°C
- Operating Humidity: 5%RH~95%RH
- Dimension: 23*24.5mm/0.91*0.96"



Board Overview



TOP



BOTTOM

No.	Silkscreen	Function
1	VIN	+
2	GND	-
3	RX	UART Receive
4	TX	UART Send
5	DACR	DAC Output
6	DACL	DAC Output
7	L+	Left channel output
8	L-	Left channel output

8	L-	Left channel output
9	R+	Right channel output
10	R-	Right channel output

No.	Silkscreen	Function
11	PLAY	Function button
12	KEY	Function button

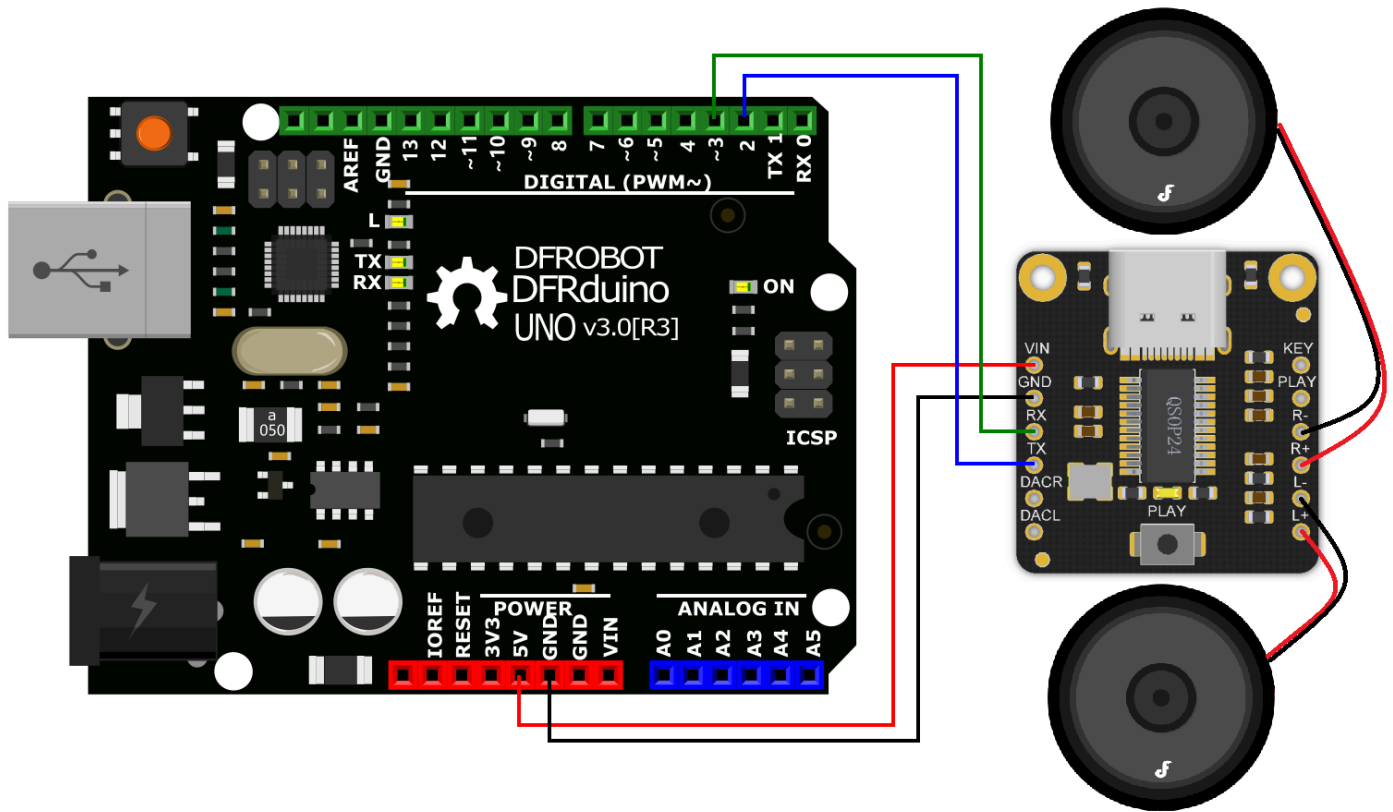
Tutorial

- Before using, please connect the module to a PC with an USB cable, and save audio files into the module storage.
- Suggest using a 4 Ω 3W speaker.

Requirements

- **Hardware**
 - DFRduino UNO R3 (<https://www.dfrobot.com/product-838.html>) (or similar) x 1
 - DFPlayer PRO x1
 - Jumper wires
- **Software**
 - Arduino IDE (<https://www.arduino.cc/en/Main/Software>)
 - Download and install the **DFRobot_DF1201S Library and Sample Program** (https://github.com/DFRobot/DFRobot_DF1201S) (About how to install the library? (<https://www.arduino.cc/en/Guide/Libraries#.UxU8mdzF9H0>))

Connection Diagram



Control by Button

1. Tap the PLAY button to play or pause(Blue LED keeps on when playing and blinks slowly when pausing).
2. Long-press the for 2 seconds to switch to the next song(Blue LED blinks twice when switching).

Control by Software

- Main API Function List

```
/**
 * @brief Set volume
 * @param vol:0-30
 * @return true or false
 */
bool setVol(uint8_t vol);

/**
 * @brief Set playback mode
 * @param ePlayMode_t:SINGLECYCLE,ALLCYCLE,SINGLE,RANDOM,FOLDER
 * @return true or false
 */
bool setPlayMode(ePlayMode_t mode);

/**
 * @brief Play
 * @return true or false
 */
bool start();

/**
 * @brief Pause
 * @return true or false
 */
bool pause();

/**
 * @brief Next
 * @return true or false
 */
bool next();

/**
 * @brief Previous
 * @return true or false
 */
```

```
bool last();

/**
 * @brief Fast forward the currently-playing song
 * @param second FF time(Unit: S)
 */
bool fastForward(uint16_t second);

/**
 * @brief Fast Rewind the currently-playing song
 * @param second FR time(Unit: S)
 */
bool fastReverse(uint16_t second);

/**
 * @brief Let the currently-playing song start playing from a particular time
 * @param second Fixed time point
 */
bool setPlayTime(uint16_t second);

/**
 * @brief Get file number
 */
uint16_t getCurFileNumber();

/**
 * @brief Get the number of files available to play
 */
uint16_t getTotalFile();

/**
 * @brief Get the time length the current song has played
 */
uint16_t getCurTime();

/**
 * @brief Get the total length of the currently-playing song
 */
uint16_t getTotalTime();

/**
 * @brief Get the name of the playing file
```

```
*/  
String  getFileName();  
  
/**  
 * @brief Play file of the specific path  
 * @param The designated path  
 */  
bool playSpecFile(String str);  
  
/**  
 * @brief Play the file of specific number, the numbers are arranged accordi  
 * @param File number, can be obtained by getCurFileNumber()  
 */  
bool playFileNum(int16_t num);  
  
/**  
 * @brief Delete the currently-playing file  
 * @return true or false  
 */  
bool delCurFile();
```

Sample Code - Play

When powering on, the module enters music mode and starts playing audio files, the current operation will be printed on the serial port.


```
/*!
 *@file play.ino
 *@brief Music Playing Example Program
 *@details Experimental phenomenon: control MP3 play music, obtain song information
 *@copyright Copyright (c) 2010 DFRobot Co.Ltd (http://www.dfrobot.com)
 *@license The MIT license (MIT)
 *@author [fengli](li.feng@dfrobot.com)
 *@version V1.1
 *@date 2021-10-15
 *@url https://github.com/DFRobot/DFRobot_DF1201S
 */

#include <DFRobot_DF1201S.h>
#include <SoftwareSerial.h>

SoftwareSerial DF1201SSerial(2, 3); //RX TX

DFRobot_DF1201S DF1201S;
void setup(void){
  Serial.begin(115200);
  DF1201SSerial.begin(115200);
  while(!DF1201S.begin(DF1201SSerial)){
    Serial.println("Init failed, please check the wire connection!");
    delay(1000);
  }
  /*Set volume to 20*/
  DF1201S.setVol(/*VOL = */20);
  Serial.print("VOL:");
  /*Get volume*/
  Serial.println(DF1201S.getVol());
  /*Enter music mode*/
  DF1201S.switchFunction(DF1201S.MUSIC);
  /*Wait for the end of the prompt tone */
  delay(2000);
  /*Set playback mode to "repeat all"*/
```

```
DF1201S.setPlayMode(DF1201S.ALLCYCLE);
Serial.print("PlayMode:");
/*Get playback mode*/
Serial.println(DF1201S.getPlayMode());

//Set baud rate to 115200(Need to power off and restart, power-down save)
//DF1201S.setBaudRate(115200);
//Turn on indicator LED (Power-down save)
//DF1201S.setLED(true);
//Turn on the prompt tone (Power-down save)
//DF1201S.setPrompt(true);
//Enable amplifier chip
//DF1201S.enableAMP();
//Disable amplifier chip
//DF1201S.disableAMP();
}

void loop(){
  Serial.println("Start playing");
  /*Start playing*/
  DF1201S.start();
  delay(3000);
  Serial.println("Pause");
  /*Pause*/
  DF1201S.pause();
  delay(3000);
  Serial.println("Next");
  /*Play the next song*/
  DF1201S.next();
  delay(3000);
  Serial.println("Previous");
  /*Play the previous song*/
  DF1201S.last();
  delay(3000);
  Serial.println("Start playing");
  //Fast forward 10S
  DF1201S.fastForward(/*FF = */10);
  //Fast Rewind 10S
  //DF1201S.fastReverse(/*FR = */10);
  //Start the song from the 10th second
  //DF1201S.setPlayTime(/*Play Time = */10);
```

```
Serial.print("File number:");
//Get file number
Serial.println(DF1201S.getCurFileNumber());

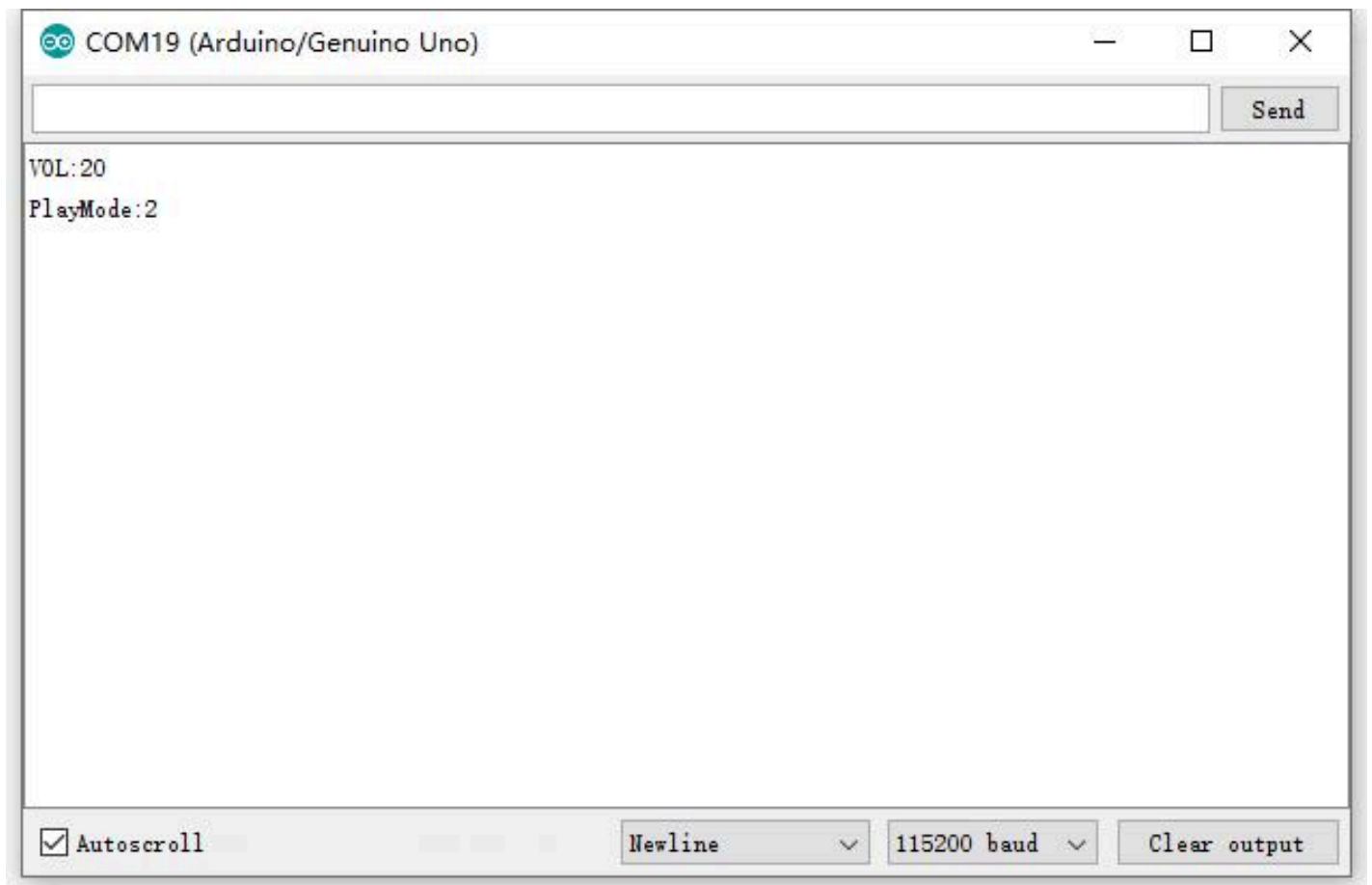
Serial.print("The number of files available to play:");
//The number of files available to play
Serial.println(DF1201S.getTotalFile());

Serial.print("The time length the current song has played:");
//Get the time length the current song has played
Serial.println(DF1201S.getCurTime());

Serial.print("The total length of the currently-playing song: ");
//Get the total length of the currently-playing song
Serial.println(DF1201S.getTotalTime());
Serial.print("The name of the currently-playing file: ");
//Get the name of the playing file
Serial.println(DF1201S.getFileName());
delay(3000);
//Play the file No.1, the numbers are arranged according to the sequence of
DF1201S.playFileNum(/*File Number = */1);
//Play the test.mp3 file in test folder
//DF1201S.playSpecFile("/test/test.mp3");

while(1);
/*Delete the currently-playing file */
//DF1201S.delCurFile();
}
```

Result



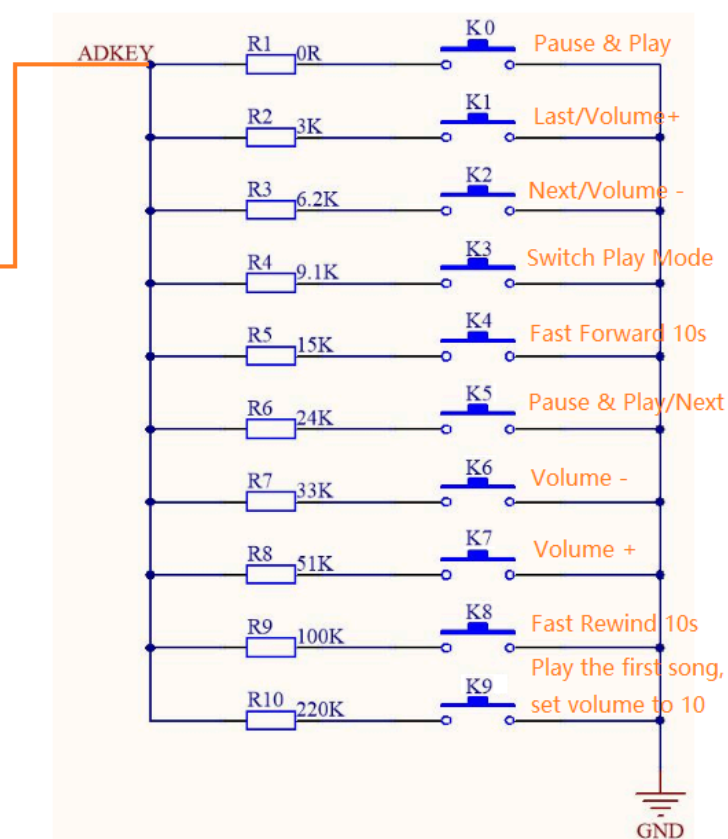
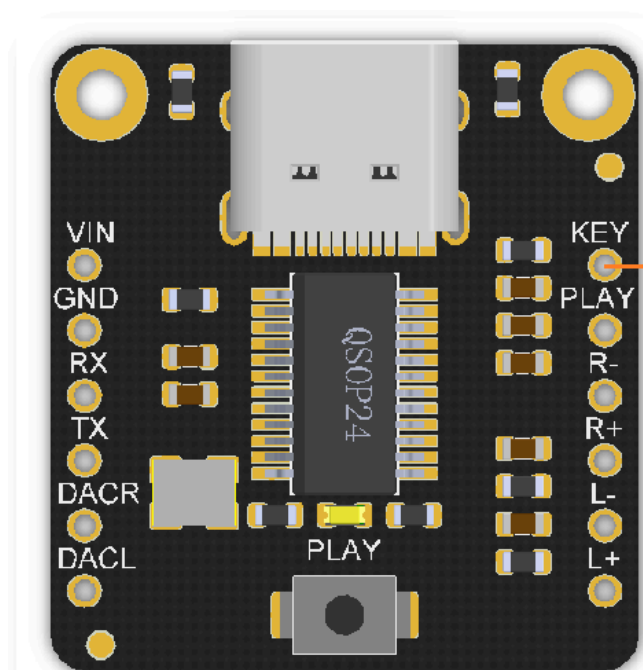
AD Button Controlling

Get **AD Button** (<https://www.dfrobot.com/product-2267.html>) from DFRobot Store or **DFRobot Distributor**. (<https://www.dfrobot.com/distributor>)

Key	Resistor in Series	Click	Long-press
K0	0R	Pause & Play	
K1	3K	Last	Volume +
K2	6.2K	Next	Volume -
K3	9.1K	Switch Play Mode	
K4	15K	Fast Forward 10s	

K5	24K	Pause & Play	Next
K6	33K	Volume -	Volume -

Key	Resistor in Series	Click	Long-press
K7	51K	Volume +	Volume +
K8	100K	Fast Rewind 10s	
K9	220K	Play the first song, set volume to 10	



AT Command Controlling

Note:

1. "\r\n" is omitted in all the commands below, and you need to add it to the end of the command in actual use. For example "AT+VOL=5\r\n", designate volume to "5".
2. Default serial baud rate: 115200

```
/**
 * @brief Test Connection
 */
AT //Test Connection

/**
 * @brief Set/Query Volume (Volume: 0-30) (Power-down save)
 * @param -n:Volume -n
 *          +n:Volume +n
 *          n:Designate volume to n
 *          ?:Query volume
 */
AT+VOL=-5 //Volume -5

AT+VOL=? //Query volume
return: VOL = [10] //The current volume is 10

/**
 * @brief Control playback mode
 * @param 1:repeat one song
 *          2:repeat all
 *          3:play one song and pause
 *          4:Play randomly
 *          5:Repeat all in the folder
 *          ?:query the current playback mode
 */
AT+PLAYMODE=1 //Switch to repeat-one-song mode

AT+PLAYMODE=? //Query current playback mode
Return: PLAYMODE =1 //Current playback mode: repeat one song

/**
 * @brief Control playing
 * @param PP:Play & Pause
 *          NEXT:next
 *          LAST:last
```

```
*/
AT+PLAY=NEXT //last

/**
 * @brief Playing time control
 * @param -n:Fast Rewind n S
 *          +n:Fast Forward n S
 *          n:Start playing from the Nth second
 */
AT+TIME=-5 //Fast Rewind 5S

/**
 * @brief Query playback information
 * @param 1:Query the file number of the currently-playing file
 *          2:Query the total number of the files
 *          3:Query the time length the song has played
 *          4:Query the total time of the currently-playing file
 *          5:Query the file name of the currently-playing file
 */
AT+QUERY=1 //Query the file number of the currently-playing file

AT+QUERY=5 //Query the file name of the currently-playing file
Return: test.mp3 //The currently-playing file is "test.mp3"

/**
 * @brief Play the file of the specified number
 * @param n:File number(Play the first file if there is no such file)
 */
AT+PLAYNUM=5 //Play file No. 5

/**
 * @brief Play the specific file
 * @param File path
 */
AT+PLAYFILE=/DF_REC/test.MP3 //Play the test.mp3 under DF_REC once

/**
 * @brief Amplifier On/OFF command
 * @param ON, OFF
 */
AT+AMP=ON //Turn on amplifier
```

```

/**
 * @brief Delete currently-playing file
 */
AT+DEL

/**
 * @brief Set baud rate (power-down save, valid after re-powering on)
 * @param 9600、19200、38400、57600、115200
 */
AT+BAUDRATE=115200 //Set baud rate to 115200

/**
 * @brief Prompt tone ON/OFF command (Power-down save)
 * @param ON, OFF
 */
AT+PROMPT=ON //Turn on prompt tone

//LED Prompt On/Off Command(Power-down save)
/**
 * @brief LED Prompt ON/OFF command (Power-down save)
 * @param ON, OFF
 */
AT+LED=ON //Turn on LED Prompt

```

FAQ

Q: What is the naming specification for FileNumber? A: The naming convention is determined by the order in which files are written to the DFPlayer PRO, as shown in the diagram below. The arrow numbers represent the write order.



If you have any questions about using this product, please check the **FAQ list** (<https://www.dfrobot.com/forum/topic/315182>) for that product for a corresponding solution. For any questions, advice or cool ideas to share, please visit the **DFRobot Forum** (<https://www.dfrobot.com/forum/>).

More Documents

- Schematics Diagram
(<https://dfimg.dfrobot.com/nobody/wiki/8d1fc472ac24ddd58479b9ba8a912a05.pdf>)
- Dimension Diagram
(<https://dfimg.dfrobot.com/nobody/wiki/b3a4c013d4b497f6e32391b3dd108344.pdf>)
- 3D Model
(<https://dfimg.dfrobot.com/nobody/wiki/7c390ce1e0e8480ad54141c873f89147.7z>)
- DF1201S Datasheet
(<https://dfimg.dfrobot.com/nobody/wiki/a6ec053c2390018d801e2ed31f0c6329.pdf>)