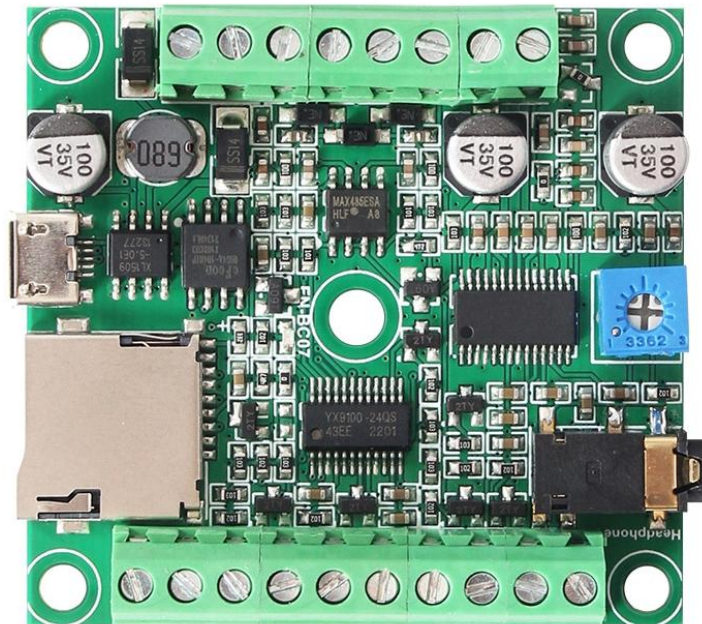


# FN-BC07 MP3 Sound Module with 15W Amplifier

## User's Manual

V1.0



## Contents

<b>1. Overviews .....</b>	<b>2</b>
1.1. Brief introduction .....	2
1.2. Features .....	2
1.3. Technical parameters .....	2
1.4. Connections .....	2
<b>2. Button Control Mode .....</b>	<b>3</b>
2.1. Set a Trigger Mode.....	3
2.2. How to Create a Config File.....	4
2.3. Audio Files Loading.....	5
<b>3. UART and RS485 Serial Control Modes .....</b>	<b>8</b>
3.1. Command Format.....	8
3.2. About Checksum .....	9
3.3. Serial Communication Commands .....	9
3.4. Detailed Annotation for Some Control Commands .....	10
3.5. Detailed Annotation for Some Query Commands.....	15



## 1. Overviews

### 1.1. Brief Introduction

FN-BC07 is a high quality MP3 sound module developed by Flyron Technology Co., Ltd. Equipped with an on board 15W amplifier, the sound module can be controlled by 7 separate buttons hooked up to the “one-on-one” inputs terminals and by RS485 or UART serial port. When it is playing a sound, it is able to drive an external load (equipment) like a warning light or a motor simultaneously. Great audio output, industry-grade design and strong anti-jamming capability make it possible to be used for many different applications.

### 1.2. Features

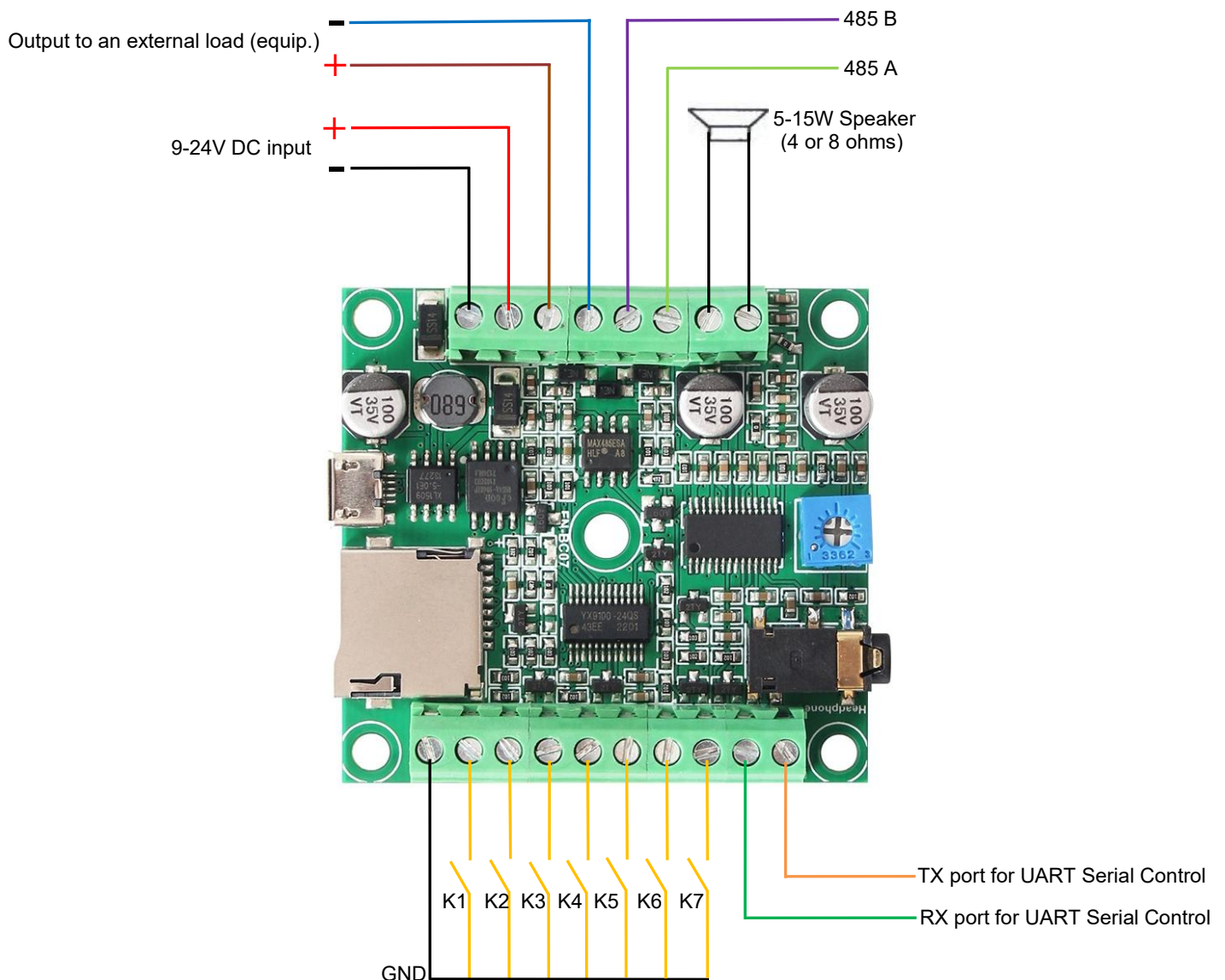
- 1). Equipped with a high quality audio decoder, which supports MP3 and WAV audio formats.
- 2). Sampling rates supported: 8/11.025/12/16/22.05/24/32/44.1/48(KHz).
- 3). 24 bit DAC output and supports dynamic range 90dB and SNR 85dB.
- 4). Supports 7 key control mode, RS485 and UART serial port control modes.
- 5). In button control mode, it supports eight different trigger modes.
  - Able to play 7 one-on-one sound files.
  - Able to work as a standard MP3 player.
  - Able to play a sound in a loop when power is applied, and any of the seven buttons can interrupt it.
  - Able to play multiple sound files per button.
  - Set a trigger mode using a config file easily.
- 6). Uses the built-in SPI flash memory and a micro SD card as the storage devices.
- 7). Built-in 8MB SPI flash memory and supports max. 32GB micro SD card.
- 8). Possible to load audio files to the flash memory directly via the micro USB connecting with computer like using a USB flash drive.
- 9). Equipped with a class D mono 15 watts amplifier that can drive a max. 15 watts speaker directly.
- 10). Equipped with a 3.5mm audio jack for stereo output that can drive an earphone directly or connect with an external amplifier.
- 11). Adjustable sound volume for speaker output through the blue potentiometer.
- 12). Able to drive an external equipment simultaneously when it is playing a sound.
- 13). Wide range for power input (DC 9-24V).
- 14). PCB size: 52mmx49mm

### 1.3. Technical Parameters

- 1). Working voltage: DC 9V-24V
- 2). Working current:  $\leq 2500\text{mA}$  (with external load)
- 3). Output current to an external load:  $\leq 1000\text{mA}$
- 4). Power consumption:  $\leq 25\text{W}$  (with external load)
- 5). On-board flash memory size: 8MB
- 6). Audio format: MP3 and WAV

### 1.4. Connections

K1, K2, K3, K4, K5, K6 and K7 are representing Normally Open (N.O.) manual buttons. TX port and RX port are used for UART serial control mode, and 485 A and 485 B are used for RS485 control mode. Please refer to the connection example below.



**Notes:**

- 1). The output ports to an external load (equip.) have the same voltage as the power supply you use, so suppose you need to connect with a 12V light, the power supply you need to use must be 12V. If you don't need to connect with an external equipment, just ignore this.
- 2). When you use buttons to control the module, please skip the connections for UART and RS485, and vice versa.

**2. Button Control Mode**  
**2.1. Select a Trigger Mode**

In button control mode, there are 8 trigger modes available for users to choose according to the actual needs. Any of these 8 trigger modes can be set/acquired through a config file, which is a text file(.txt). Users just need to fill in a number that is corresponding to a trigger mode in a new built text file. Save it and rename the file "Config" or any name you like, then put it in the root directory of the micro SD card or the flash memory together with the audio files or folders. Please refer to the below sheet about the number and the associated trigger modes.

Number in Config File	Corresponding Trigger Mode
0	Short press the button to start playing, and during playing, if you press the button again, the playback will be interruptible and it will play from the beginning.
1	Press the button and hold to start playing repeatedly, and when the button is released the board stops playing.
2	Short press the button to start playing, and during playing, if you press the button again, the playback will be NOT interruptible.
3	In this mode, K1 works as Next, K2 Previous, K3 Play/Pause, K4 Stop, K5 Volume Up, K6 Volume Down, K7 Random playback (it plays a sound randomly after each triggering). Special function: When K1 is short-circuited to GND first and use this mode in the config file and once power is applied, the board is able to play a sound in a loop or play multiple sounds one by one in a loop until power is off.
4	In this mode, sound file 000.mp3 will play in a loop when power is applied. Any button from K1 to K7 (associated 001.mp3 to 007.mp3) can interrupt it and switch to play the associated sound file. Once the associated sound file finishes the playback, sound file 000.mp3 will continue to play in a loop.
5	Each button from K1 to K7 have their associated folders from 01 to 07 respectively. Each folder can store multiple files (for example from 001.mp3 to 030.mp3). Short press the button to play sound file 001.mp3, and short press again to play the next file 002.mp3, and so on. When the last sound file finish playing, short press again to go back to playing file 001.mp3. During playing, if you press the button again, the playback will be interruptible and it will play next file.
6	The same as mode "5", but during playing, the playback will be uninterruptible.
7	Each button from K1 to K7 have their associated folders from 01 to 07 respectively. Each folder can store multiple files (for example from 001.mp3 to 030.mp3). Short press the button to play all of the sound files in order one by one. During playing, if you press the button again, the playback will be interruptible and it will play next file and continue to play the rest of the file. When the last sound file finishes playback, it'll stop.

**Note: If there is no config file on the memory/micro SD card, the board will always works with the mode "0" by default.**

## 2.2. How to create a config file

Users can create a config file without or with volume setting. The volume potentiometer on the board controls the speaker output only, so if some users want to adjust volume from the 3.5mm audio jack output, it's necessary to create a config file with volume setting, otherwise a config file without volume setting is enough. By the way, in the same time the speaker output is also subject to the volume setting in the config file.

### 2.2.1. Create a config file without volume setting

- 1). Firstly create a new text file on computer (desktop or somewhere else).
- 2). Open it and enter a number (mode) you need. Suppose you need mode "1", just enter "1". See below.



- 3). Save it and close it.
- 4). Rename the file "Config" or any other name you like.

### 2.2.2. Create a config file with volume setting

There're thirty-one volume levels from "00" to "30". "00" means mute while "30" means the max. volume level.

- 1). Firstly create a new text file on computer (desktop or somewhere else).
- 2). Open it and enter a number (mode) you need, and enter a volume level right after the mode number. See below.



- 3). Save it and close it.
- 4). Rename the file "Config" or any other name you like.

## 2.3. Audio Files Loading/Updating

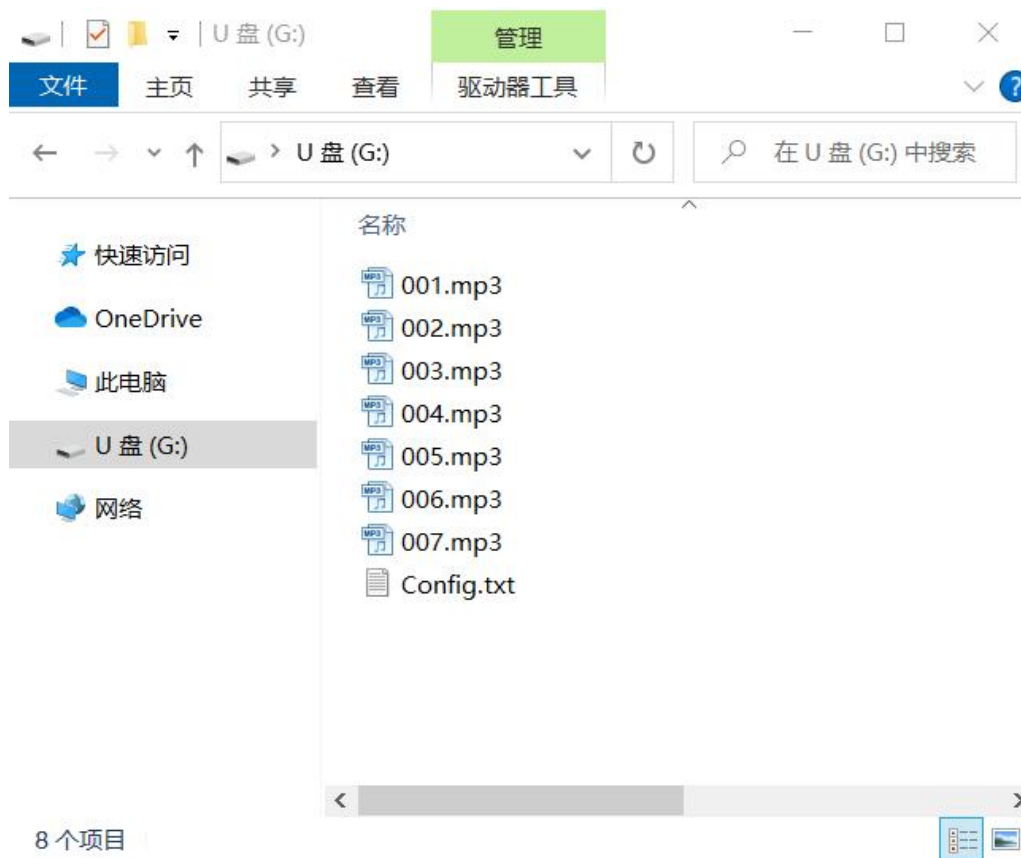
Users can use a micro USB data cable to connect the board to computer. The built-in flash memory will be detected as a USB flash drive on computer. If the built-in flash memory is not large enough to store your audio file, you can use a micro SD card instead. When there is an inserted micro SD card on the board, there will be two simulated USB flash drives on computer (one is the built-in flash memory and the other one is the micro SD card), so please note to recognize.

### 2.3.1. For Trigger Mode 0-3

To these four modes, the audio files need to be placed on the root directory of the storage device (built-in flash memory or micro SD card). The arrangements of the audio files are managed by a physical index order. In other words, the file that is to be loaded first in the storage device will be associated with input "K1". The last file to be loaded in the storage device will be associated with input "K7". In order to guarantee a correct 'one-on-one' order, please refer to the following steps.

- 1). Build a new folder on the computer and put the 7 audio files in this new folder.
- 2). Rename the audio files from 001.mp3 to "007.mp3", and make sure they are ranked from "001.mp3" to "007.mp3" in order.
- 3). Connect the module to computer through a USB data cable.
- 4). Delete the pre-loaded audio files for tested purpose at factory, or empty the micro SD card if it has other files inside.
- 5). Go back to the folder and select all of the 7 audio files in the folder.
- 6). Right click on the first file (001.mp3) and choose "Send to removable disk" or "Send to USB flash drive".
- 7). This will send the 7 audio files to the on-board flash memory or micro SD card one by one in a correct sequence.

8). Move the prepared config file with the trigger mode needed onto the root directory together with the audio files as below then refresh.



9). Safely remove the USB connection from computer.

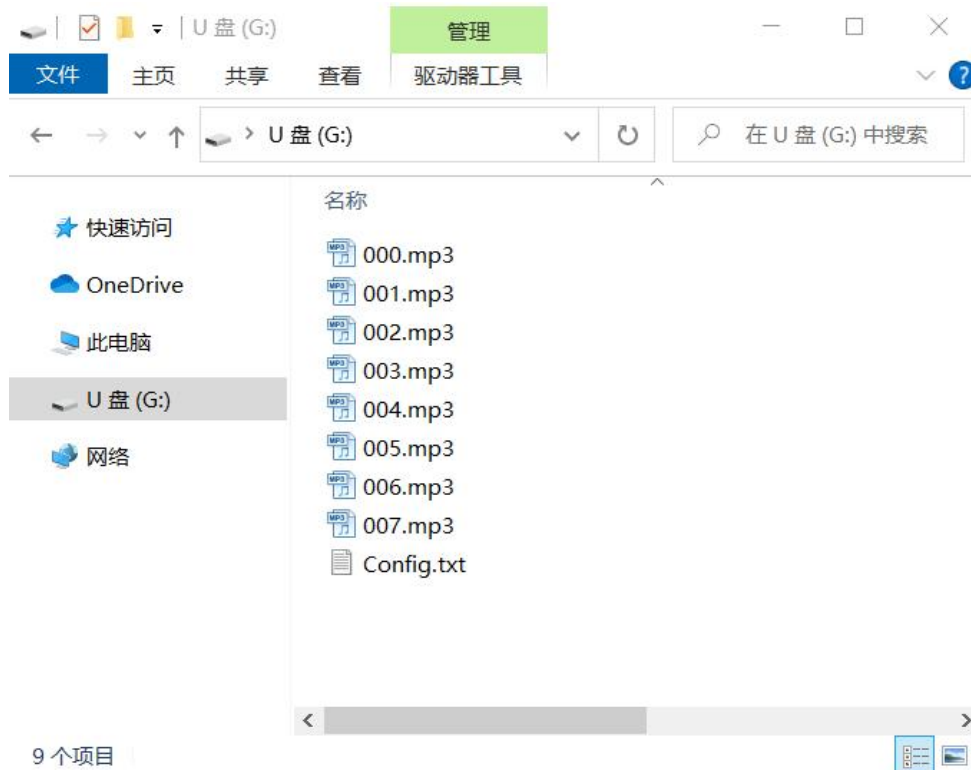
**Notes: 1). When you use the trigger mode "3", you can definitely place more than 7 audio files.**

**2). When a micro SD card is plugged into the module, only audio files from the SD card will be played.**

### 2.3.2. For Trigger Mode 4

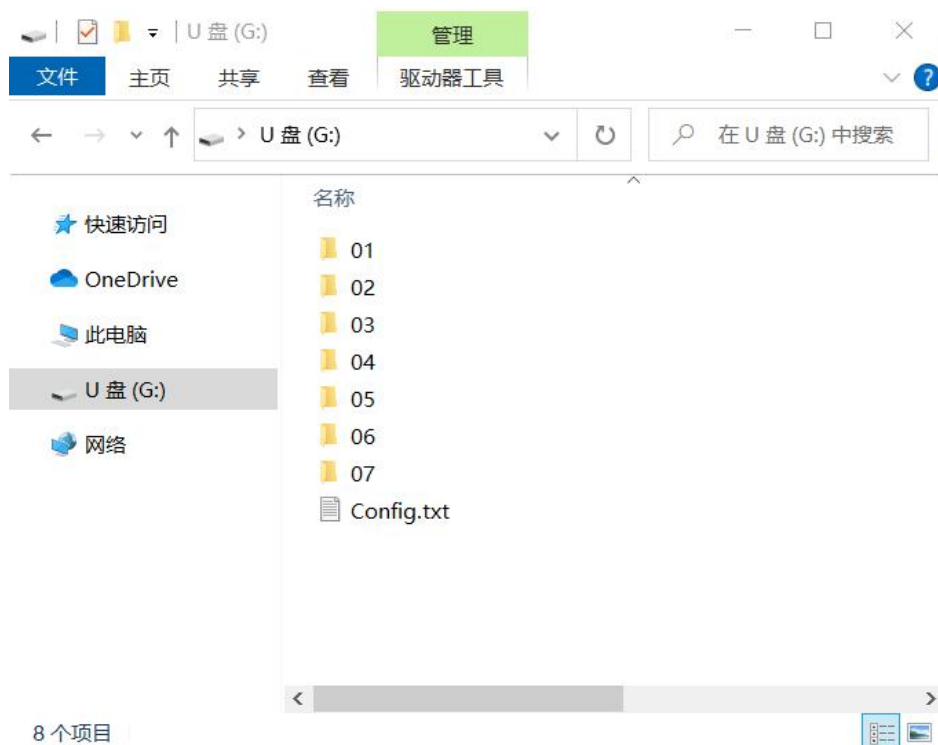
Like the first four trigger modes, the audio files need to be placed on the root directory of the storage device (built-in flash memory or micro SD card). In this mode, the module recognizes the files by name instead of physical index order, so it'll be alright to directly copy your audio files onto the storage device and rename the files 000.mp3-007.mp3 according to your actual needs. Please refer to the screenshot below.



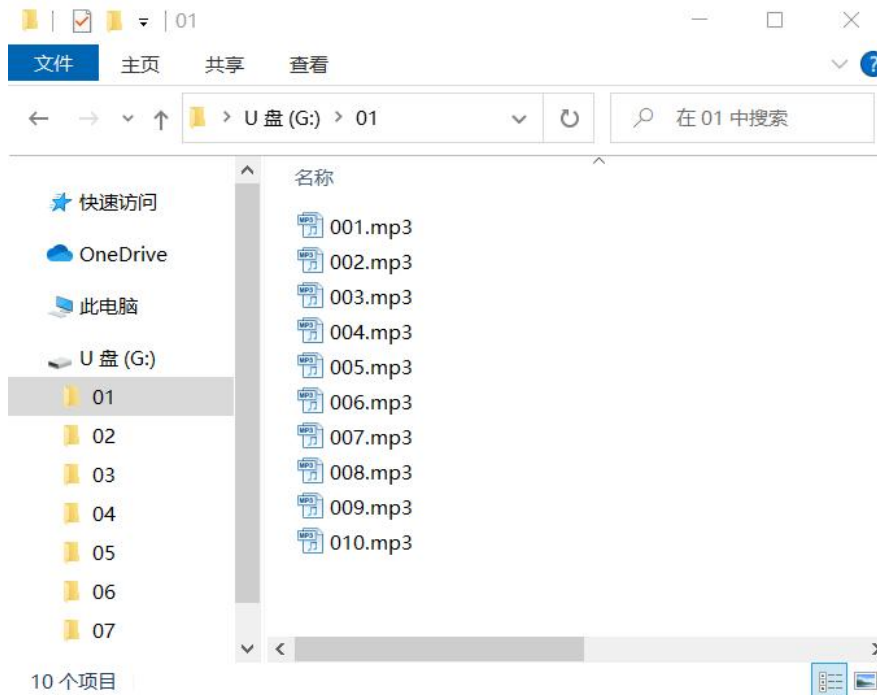


### 2.3.3. For Trigger Mode 5-7

At first, users need to create seven folders on the storage device (built-in flash memory or micro SD card), and rename them 01, 02, 03, till 07, then put the config file together with the folders on the root directory. Of course, if you don't have to use so many buttons, then you don't need to create as many as seven folders. See the screenshot as below.



After that, copy your audio files to each associated folder, and rename them 001.mp3, 002.mp3, and so on. In these three modes, the module recognize the audio files by physical index order, so when you copy audio files to each folder, please note the copy order. See the screenshot as below.



**Note: When you rename a file, you can still keep the original file name and you can rename it like 001-Never Say Goodbye.mp3, 002-Season in the Sun.mp3, 003-Angel.mp3, and so on. This rule applies to all of the trigger modes.**

### 3. UART and RS485 Serial Control Modes

In the fields of industrial control, power communication, smart instruments, etc., serial communication, as a common communication in the field of control, is usually used for data exchange. On this basis, we have carried out industrial-level optimization and added frame verification, retransmission, error handling and other measures, greatly enhanced the stability and reliability of communication, and can expand more powerful RS485 networking functions. This sound module supports UART and RS485 serial communications, in order to facilitate users to do more function extensions. The default baud rate is 9600.

#### 3.1. Command Format

- Communication baud rate: 9600 bps
- Data bits: 1
- Checkout: none
- Flow Control: none

Format:	\$\$ / Ver. / Number / Command / Feedback / Param_MSB / Param_LSB / Check_MSB / Check_LSB / \$O
\$\$	Start byte 0x7E
Ver.	Version byte, 0xFF by default
Number	Number of bytes from version info to Check_LSB, typically 0x06 (checks um not counted)
Command	Command byte





Feedback	0x01: Need feedback --send confirmation back to MCU; 0x00: No need feedback
Param_MSB	Most significant byte of parameter
Param_LSB	Least significant byte of parameter
Check_MSB	Most significant byte of checksum
Check_LSB	Least significant byte of checksum
\$O	End byte 0xEF

For example, if we specify playback of SD card, we need to send the command "7E FF 06 09 00 00 02 FF F0 EF". The number is 6 bytes, and these 6 bytes are "FF 06 09 00 00 02". Start byte, end byte and checksum are not counted.

### 3.2. About Checksum

Regarding to calculating checksum, you can use the following formula to count.

$$\text{Checksum (2 bytes)} = 0xFFFF - (\text{CMD} + \text{Feedback} + \text{Para\_MSB} + \text{Para\_LSB}) + 1$$

Normally it's okay whether users choose to use checksum or not, our module can receive a serial data with or without checksum, but some of users use a MCU without crystal oscillator, so if in that case we strongly suggest users to add checksum to make sure the communication stability.

### 3.3. Serial Communication Commands

#### 3.3.1. Control commands

Command	Description	Serial Command [with checksum]	Serial Command [without checksum]	Note
0x01	Play Next	7E FF 06 01 00 00 00 FE FA EF	7E FF 06 01 00 00 00 EF	
0x02	Play Previous	7E FF 06 02 00 00 00 FE F9 EF	7E FF 06 02 00 00 00 EF	
0x03	Specify playback of a track in the root directory	7E FF 06 03 00 00 01 FE F7 EF	7E FF 06 03 00 00 01 EF	Specify playback of the 1st track
		7E FF 06 03 00 00 02 FE F6 EF	7E FF 06 03 00 00 02 EF	Specify playback of the 2nd track
		7E FF 06 03 00 00 0A FE EE EF	7E FF 06 03 00 00 0A EF	Specify playback of the 10th track
0x04	Increase volume	7E FF 06 04 00 00 00 FE F7 EF	7E FF 06 04 00 00 00 EF	
0x05	Decrease volume	7E FF 06 05 00 00 00 FE F6 EF	7E FF 06 05 00 00 00 EF	
0x06	Specify volume	7E FF 06 06 00 00 1E FE D7 EF	7E FF 06 06 00 00 1E EF	Specified volume is level 30
0x08	Specify single repeat playback in the root directory	7E FF 06 08 00 00 01 FE F2 EF	7E FF 06 08 00 00 01 EF	Repeatedly play the 1st track
		7E FF 06 08 00 00 02 FE F1 EF	7E FF 06 08 00 00 02 EF	Repeatedly play the 2nd track
		7E FF 06 08 00 00 0A FE E9 EF	7E FF 06 08 00 00 0A EF	Repeatedly play the 10th track
0x09	Specify playback of a device	7E FF 06 09 00 00 02 FE F0 EF	7E FF 06 09 00 00 02 EF	Specified device is SD card
		7E FF 06 09 00 00 04 FE ED EF	7E FF 06 09 00 00 04 EF	Specified device is built-in SPI flash
0x0B	Set baud rate	7E FF 06 0B 00 00 01 FE EF EF	7E FF 06 0B 00 00 01 EF	Baud rate is 9600
		7E FF 06 0B 00 00 02 FE EE EF	7E FF 06 0B 00 00 02 EF	Baud rate is 19200



0x0C	Reset	7E FF 06 0C 00 00 00 FE EF EF	7E FF 06 0C 00 00 00 EF	
0x0D	Play	7E FF 06 0D 00 00 00 FE EE EF	7E FF 06 0D 00 00 00 EF	
0x0E	Pause	7E FF 06 0E 00 00 00 FE ED EF	7E FF 06 0E 00 00 00 EF	
0x0F	Specify playback of a track in a folder	7E FF 06 0F 00 01 01 FE EA EF	7E FF 06 0F 00 01 01 EF	Specify track "001" in the folder "01"
		7E FF 06 0F 00 01 02 FE E9 EF	7E FF 06 0F 00 01 02 EF	Specify track "002" in the folder "01"
0x11	Play all tracks in a loop	7E FF 06 11 00 00 01 FE E9 EF	7E FF 06 11 00 00 01 EF	Start playing all tracks in a loop
		7E FF 06 11 00 00 00 FE EA EF	7E FF 06 11 00 00 00 EF	Stop playing all tracks in a loop
0x16	Stop	7E FF 06 16 00 00 00 FE E5 EF	7E FF 06 16 00 00 00 EF	Stop all playback tasks
0x17	Specify repeat playback of a folder	7E FF 06 17 00 02 00 FE E2 EF	7E FF 06 17 00 02 00 EF	Specify repeat playback of the folder "02"
		7E FF 06 17 00 01 00 FE E3 EF	7E FF 06 17 00 01 00 EF	Specify repeat playback of the folder "01"
0x18	Set random playback	7E FF 06 18 00 00 00 FE E3 EF	7E FF 06 18 00 00 00 EF	Random playback of the whole device
0x19	Set repeat playback of current track	7E FF 06 19 00 00 00 FE E2 EF	7E FF 06 19 00 00 00 EF	Turn on single repeat playback
		7E FF 06 19 00 00 01 FE E1 EF	7E FF 06 19 00 00 01 EF	Turn off single repeat playback
0x21	Combination playback			
0x25	Insert an advertisement			
0xC0	Set an address (for RS485 only)	7E FF 06 C0 00 00 01 FE 3A EF	7E FF 06 C0 00 00 01 EF	
		7E FF 06 C0 00 00 63 FD D8 EF	7E FF 06 C0 00 00 63 EF	

### 3.3.2. Query commands

Command	Description	Serial Command [with c heck sum]	Serial Command [without c heck sum]	Note
0x3F	Query current online storage device	7E FF 06 3F 00 00 00 FE BC EF	7E FF 06 3F 00 00 00 EF	
0x42	Query current status	7E FF 06 42 00 00 00 FE B9 EF	7E FF 06 42 00 00 00 EF	
0x43	Query current volume	7E FF 06 43 00 00 00 FE B8 EF	7E FF 06 43 00 00 00 EF	
0x48	Query number of tracks in the micro SD card	7E FF 06 48 00 00 00 FE B3 EF	7E FF 06 48 00 00 00 EF	Total file numbers of current device
0x49	Query number of tracks in the SPI flash	7E FF 06 49 00 00 00 FE B2 EF	7E FF 06 49 00 00 00 EF	Total file numbers of current device
0x4C	Query current track in the micro SD card	7E FF 06 4C 00 00 00 FE AF EF	7E FF 06 4C 00 00 00 EF	Query the track being played
0x4D	Query current track in the SPI flash	7E FF 06 4D 00 00 00 FE AE EF	7E FF 06 4D 00 00 00 EF	Query the track being played
0x4E	Query number of tracks in a folder	7E FF 06 4E 00 00 01 FE AC EF	7E FF 06 4E 00 01 00 EF	
0x4F	Query number of folders in the current storage device	7E FF 06 4F 00 00 00 FE AC EF	7E FF 06 4F 00 00 00 EF	

### 3.4. Detailed Annotation for Some Control Commands

#### 3.4.1. Specify playback of a track in the root directory of the storage device (0x03)



The available selectable tracks are from 1<sup>st</sup> to 3000<sup>th</sup> in the root directory of the storage device. Actually it can support more, but if we make it support more, the operation speed will become slow. Usually most of applications do not need to support much more files. Normally if the total track quantity is less than 1,000 in the storage device the response speed of the module is quicker than 50ms. If the total track quantity is more than 1000 the response speed is a little slower than 100ms.

1) For example, if set the first track to be played, send the command 7E FF 06 03 00 00 01 FE F7 EF

- 7E --- Start byte
- FF --- Version Information
- 06 --- Number of bytes
- 03 --- Actual command (specify playback of a track)
- 00 --- 0 x01 : need feedback ; 0 x00 : no need feedback
- 00 --- Most significant byte of the track (MSB of Parameter)
- 01 --- Least significant byte of the track (LSB of Parameter)
- FE --- Most significant byte of checksum (MSB of checksum)
- F7 --- Least significant byte of checksum (LSB of checksum )
- EF --- End byte 0xEF

2) Regarding track selection, if the 100<sup>th</sup> song (track) is selected to be played, firstly convert 100 to hexadecimal. It is double-byte by default, i.e. 0 x0064. MS B=0x00 ; LS B=0x64

### 3.4.2. Specify volume (0x06)

- 1) Our system power-on default volume is level 30 (max.), if you want to set the volume, then directly send the corresponding command.
- 2) For example, if you specify the volume to level 15, send the command 7E FF 06 06 00 00 0F FF D5 EF.
- 3) MSB=0x00; LSB=0x0F, 15 is converted to hexadecimal "0x000F".
- 4) The volume set by the command will be reset when the device is restarted. If you need the power-off memory, you can set the volume through a config file.

### 3.4.3. Specify single repeat playback in the root directory of the storage device (0x08)

Repeatedly play 1 <sup>st</sup> track	7E FF 06 08 00 00 01 xx xx EF
Repeatedly play 2 <sup>nd</sup> track	7E FF 06 08 00 00 02 xx xx EF
Repeatedly play 3 <sup>rd</sup> track	7E FF 06 08 00 00 03 xx xx EF

During single repeat playback, you can still normally execute the operations Play/Pause, Previous, Next, Volume +/- and repeat playback. Users can specify single track playback or send the stop command to turn off single repeat playback status.

### 3.4.4 Specify playback of a device (0x09)

Specify playback of SD card	7E FF 06 09 00 00 02 FE F0 EF
-----------------------------	-------------------------------

Specify playback of SPI flash	7E FF 06 09 00 00 04 FE EE EF
-------------------------------	-------------------------------

- 1) The module supports two types of playback devices (micro SD card and on-board SPI flash).
- 2) It automatically enters into the standby status after specifying a device, waiting for MCU to specify a track to play. It takes about 200ms from specifying device to the module finishes initialization of file system. Please wait for 200ms and then send the specified command to play a track.

**3.4.5. Set baud rate (0x0B)**

Set baud rate to 9600	7E FF 06 0B 00 00 <b>01</b> EF
Set baud rate to 19200	7E FF 06 0B 00 00 <b>02</b> EF

Descriptions for baud rate parameters:

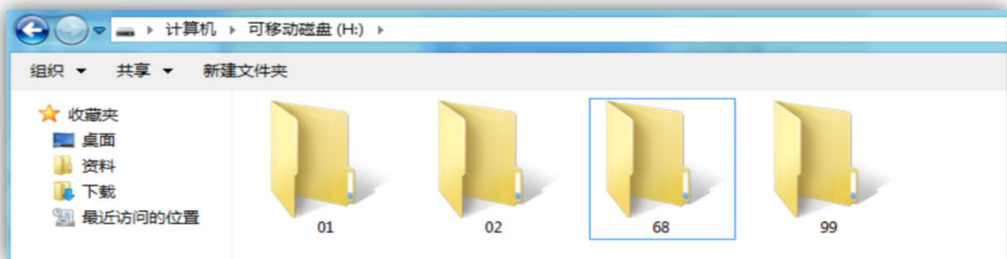
Parameter	Corresponding Baud Rate	Parameter	Corresponding Baud Rate
<b>01</b>	9600	<b>06</b>	256000
<b>02</b>	19200	<b>07</b>	35250
<b>03</b>	38400	<b>08</b>	2400
<b>04</b>	57600	<b>09</b>	4800
<b>05</b>	115200		

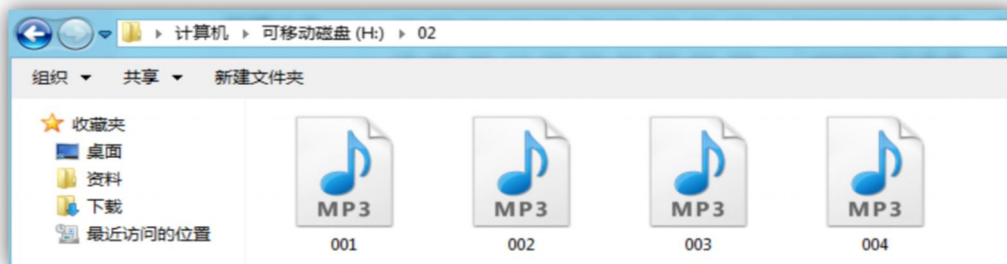
- 1) The default baud rate of the device is 9600. After the baud rate is set, the chip will memorize it, and the baud rate will become the set baud rate after the device is re-powered on.
- 2) After setting the baud rate, please wait for 1 second, and then send the reset command [0x0C], or power off and restart to take effect.

**3.4.6. Specify playback of a track in a folder (0x0F)**

Specify playback of track 001 in the folder 01	7E FF 06 0F 00 01 01 xx xx EF
Specify playback of track 100 in the folder 11	7E FF 06 0F 00 0B 64 xx xx EF
Specify playback of track 255 in the folder 99	7E FF 06 0F 00 63 FF xx xx EF

- 1) The default folders are named as "01", "11", "99" in this way. In order to be with a better system stability, it is made to support maximum 99 folders and maximum 255 tracks in each folder..
- 2) For example, if specify to play "100.mp3" in the folder "01", send the command 7E FF 06 0F 00 01 64 xx xx EF.  
MSB: represents the name of the folder, maximum supports 99 folders from 01 - 99.  
LSB: represents the track, maximum supports 255 tracks from "0x01" to "0xFF".
- 3) You must specify both the folder and the file name to target a track. This feature supports MP3 and WAV audio formats.
- 4) The following two images illustrating the naming method of folders and files.





### 3.4.7. Specify repeat playback of a folder (0x17)

Specify repeat playback of the folder "02"	7E FF 06 17 00 00 02 FE E2 EF
Specify re pea t playback of the folder "01"	7E FF 06 17 00 00 01 FE E3 EF

The folder names must be 01- 99, and no more than 99. After sending the command, it repeatedly plays the tracks in the specific folder and it will not stop until it receives a command to stop.

### 3.4.8. Set random playback (0x18)

Random playback of the whole storage device	7E FF 06 18 00 00 00 FE E3 EF
---	-------------------------------

This command is used to randomly play all of the tracks in the storage device according to physical sequence and no matter if there is a folder or not in the device.

### 3.4.9. Set repeat playback of current track (0x19)

Turn on single repeat playback	7E FF 06 19 00 00 00 FE E2 EF
Turn off single repeat playback	7E FF 06 19 00 00 01 FE E1 EF

During playback, send the turn-on command, and it will repeatedly play the current track. If the module is at Pause or Stop status, it will not respond to this command. If you need to turn off repeat playback, just send the turn-off command.

### 3.4.10. Combination playback (0x21)

Combination of 3 tracks	7E FF 09 21 01 02 02 03 01 04 EF	Play track 002 in folder 01, track 003 in folder 02 and track 004 in folder
Combination of 9 tracks with checksum	7E FF 15 21 01 02 02 03 01 04 01 03 01 04 01 05 02 08 03 04 03 01 FE 9A EF	Play track 002 in folder 01, track 003 in folder 02, track 004 in folder 01, track 003 in folder 01, track 004 in folder 01, track 005 in folder 01, track 008 in folder 02, track 004 in folder 03, and track 003 in folder 03.
Combination of 9 tracks without checksum	7E FF 15 21 01 02 02 03 01 04 01 03 01 04 01 05 02 08 03 04 03 01 EF	Play track 002 in folder 01, track 003 in folder 02, track 004 in folder 01, track 003 in folder 01, track 004 in folder 01, track 005 in folder 01, track 008 in folder 02, track 004 in folder 03, and track 003 in folder 03.



1) . We added this function to meet some users' special needs that when users need to send only one frame data to play multiple tracks one by one without pause. It supports maximum 30 tracks together for combination playback. All of the sound files used for combination playback need to be put in folders (folder 01-folder 99).

2). If MCU/controller sends a frame data as **7E FF 15 21 01 02 01 03 01 04 01 05 01 06 02 01 03 05 04 07 05 09 EF**, see the analysis as below.

Command: 0x21

Number of bytes: 0x15=21 bytes --- **FF 15 21 01 02 01 03 01 04 01 05 01 06 02 01 03 05 04 07 05 09** (two parameters for one track, i.e. the folder number and the track number)

The module will play track 002 in folder 01, track 003 in folder 01, track 004 in folder 01, track 005 in folder 01, track 006 in folder 01, track 001 in folder 02, track 005 in folder 03, track 007 in folder 04, and track 009 in folder 05.

3). During combination playback, it is allowed to Play/Pause and set volume, but not allowed to set Previous and Next. If need to stop, just direct send the stop command. And it is not allowed to play another group of combination during it is working. Users need to send the stop command to stop the current combination playback before start another group of combination playback.

4). If a track specified to be played in combination is not in the folder, it will stop playing at this track position, so please make sure the track specified to play must be available in the folder.

5). If users are very strict to the combination playback, please edit the sound sources with some audio edit software like Adobe Audition or GoldWave to cut off the silence at the beginning and the end of the sound.

6). Because this string of the control command is very long, we removed the bytes of "Feedback".

**3.4.11. Insert an advertisement (0x25)**

Insert track "001" in the folder "ADVERT1"	7E FF 06 25 00 01 01 FE D4 EF
Insert track "002" in the folder "ADVERT1"	7E FF 06 25 00 01 02 FE D3 EF
Insert track "001" in the folder "ADVERT2"	7E FF 06 25 00 02 01 FE D3 EF

- 1) This module supports insert advertisements (inter-cut) during playback of a track, so that it can meet some special needs for some applications.
- 2) After sending the command "0x25", the system will save the ID V3 information of the track being played and pause, and then it will play the specified insert track (advertisement). When the insert track is finished, the system will go back and continue to play the track that was interrupted until to the end.
- 3) The setting method is to build a folder named "ADVERT1" in the storage device and put the tracks (ads) you need in the folder and name the files as "001 .mp3/wav", 002.mp3/wav. It supports maximum 9 folders from "ADVERT1 to ADVERT9", and each folder can have maximum 255 tracks.
- 4) If you send an insert command when the module is at Pause status or Stop status, it will not work and there will be returned error information. In the course of an inter-cut, you can continue to insert the other tracks (ads). When the last inserted track goes to the end, the systems till goes back to the ID V3 position saved at the first time.

**3.4.12 Set an address (0xC0)**

This function is used for RS485 control mode only. With this function, it is equivalent to giving each device a unique name, so



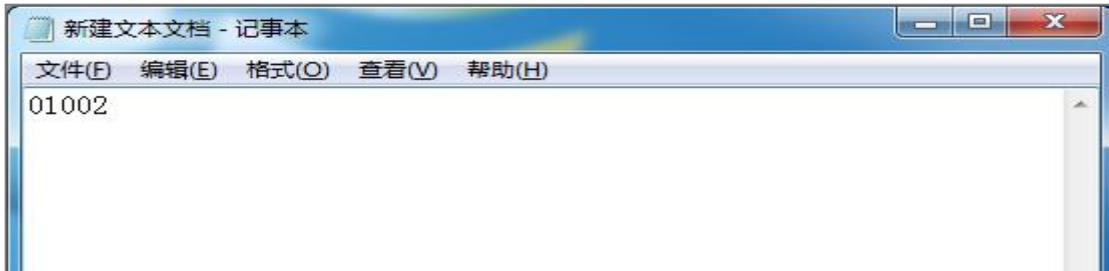
multiple devices can be controlled individually via the RS485 bus.

It supports two ways to setting an address to the module as below.

- 1). Set an address through the serial command

Set the address to 01	7E FF 06 C0 00 00 <b>01</b> EF
Set the address to 99	7E FF 06 C0 00 00 <b>63</b> EF

- 2). Set an address through a config file



There is a 5-digit value in the config file. The 1<sup>st</sup> digit represents the function of the buttons; the 2<sup>nd</sup> and 3<sup>rd</sup> digits represent the volume setting; the 4<sup>th</sup> and 5<sup>th</sup> digits are the device address. Let's take "01002" as an example to further explain as below.

The 1<sup>st</sup> digit "0" represents the function of the buttons.

The 2<sup>nd</sup> and 3<sup>rd</sup> digits "10" represents the volume level.

The 4<sup>th</sup> and 5<sup>th</sup> digits "02" represents the address of the device.

If the address was set to "01", when you send the serial command from RS485 bus to control this module, "0xFF" will be changed to "0x01", like "7E **01** 06 01 00 00 00 EF". If the address was set to "99", when you send the serial command from RS485 bus to control this module, "0xFF" will be changed to "0x63", like "7E **63** 06 01 00 00 00 EF".

There are two types of addresses here. One of the two types of addresses is like this, and it can be set as many as to 99. This number is written in decimal, and the returned address from the chip of the module will be "0x63" in hex. When setting the address, be sure not to exceed 99. The other one is the super address "0xFF". No matter what address is set to through sending the serial command or using the config file, this command is always valid.

- 3). After the address is set by the command, the address will take effect immediately with power-off memory.
- 4). If the address is already set in the config file, the address set by the command will not be remembered after the power is turned off, and the address set by the config file will take priority after restarting.

### 3.5. Detailed Annotation for Some Query Commands

#### 3.5.1. Query current online storage device

Query current online storage device	7E FF 06 3F 00 00 00 FE BC EF
-------------------------------------	-------------------------------

When the module is working, users can use the command as above (0x3F) to query the status of the online storage devices. For example, if the module returns the data 7E FF 06 3F 00 00 0A xx xx EF, LSB 0x0A (0000 1010) represents SD card online. If LSB is 0x1F (0000 1111), it represents both of SD card and PC online (PC online means module is connecting with PC via a USB cable).

**3.5.2. Query current status**

Query current online storage device	7E FF 06 42 00 00 00 FE B9 EF
-------------------------------------	-------------------------------

1). There are 4 status (playing, paused playing, stopped playing and in sleep) that can be queried during the module is decoding. Users can query the current status via sending the command as above (0 x42)

2). Interpretation of returned data

Returned Data	Status
7E FF 06 42 00 02 01 xx xx EF	SD card is being played
7E FF 06 42 00 02 02 xx xx EF	SD card is paused playing
7E FF 06 42 00 04 00 xx xx EF	SPI flash is stopped playing
7E FF 06 42 00 04 01 xx xx EF	SPI flash is being played
7E FF 06 42 00 10 00 xx xx EF	Module in sleep

3). MSB and LSB Representations

MSB Representation		LSB Representation	
0x02	SD card	0x00	Stopped
0x04	SPI flash	0x01	Playing
0x10	Module in sleep mode	0x02	Paused

**3.5.3. Query number of tracks in a folder**

Query number of tracks in folder 01	7E FF 06 4E 00 00 01 FE AC EF
Query number of tracks in folder 11	7E FF 06 4E 00 00 0B FE A2 EF

If the folder queried is empty without any files, the module will report an error and the data 7E FF 06 40 00 00 06 FE B5 EF will be returned.

**3.5.4. Query number of total folders in current storage device**

Query number of total folders in current storage device	7E FF 06 4F 00 00 00 FE AC EF
---	-------------------------------

Users can query this through sending the command above. This just supports to query the folder numbers in the root directory of the device. Not possible to query the sub-folder numbers (Please do not build any sub-folders in a folder).