Modbus RTU Audible and Visual Alarm with 3 Color Signal Light for Workshop Machines and Industrial Equipment

Model No.: FN-VA603

Version: V1.0

1. Brief Introduction

FN-VA603 Modbus RTU audible and visual alarm (sound and light alarm) with 3-color signal light, launched by Flyron Technology Co., Ltd., has the characteristics of low power consumption, long life, flexible installation, and convenience of use. It's easy to replace audio files via USB connection to computer. This product adopts Modbus RTU standard communication protocol based on RS485 serial communication. According to actual needs, different light colors and flashing modes combined with corresponding warning voices achieve warning and alarm functions. Except for workshop machines and all kinds of industrial equipment, this device also can be used for all fire alarm control systems, security monitoring alarm systems and other alarm systems.

1.1. Features

- ✧ Adopts Modbus RTU standard communication protocol.
- ✧ Equipped with 3 color (red/yellow/green) LED signal light.
- ✧ Supports 3 types of flash modes (quick flash/slow flash/always on).
- ✧ Free to set the signal light to work in any color and any flash mode above along with the sound alarming.
- ✧ Able to realize sound and light alarming simultaneously.
- ✧ Able to realize sound alarming individually.
- ✧ Able to control working status of the signal light during sound alarming.
- ✧ Built-in 4MB flash memory and supports max. 4 minutes long MP3 files of 128Kbps.
- ✧ Able to replace audio files via USB connection to computer.

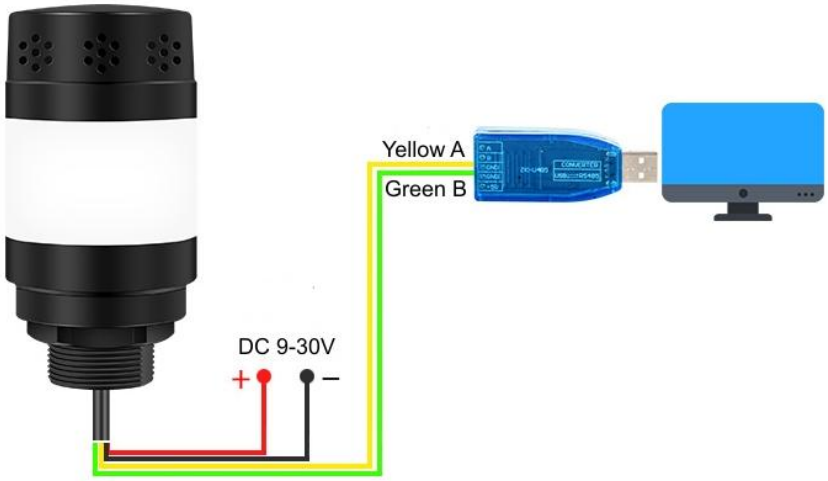
  -The internal memory of the device will be directly detected as a USB flash drive on computer. No need any auxiliary software.
- ✧ Able to set different communication baud rates (4800/9600/19200/38400/57600/115200/256000/35250).
- ✧ Able to adjust sound volume through sending the related serial commands.
- ✧ Simple to install and easy to use.

1.2. Technical parameters

Operating voltage	DC 9V-30V	Light color	red/yellow/green
Output Power	1W	Flash mode	quick flash/slow flash/always on
Audio format	MP3 (bit rate: ≤192Kbps)	Frequency response	70Hz-13KHz
Memory Capacity	4MB	Sound intensity	≤100dB

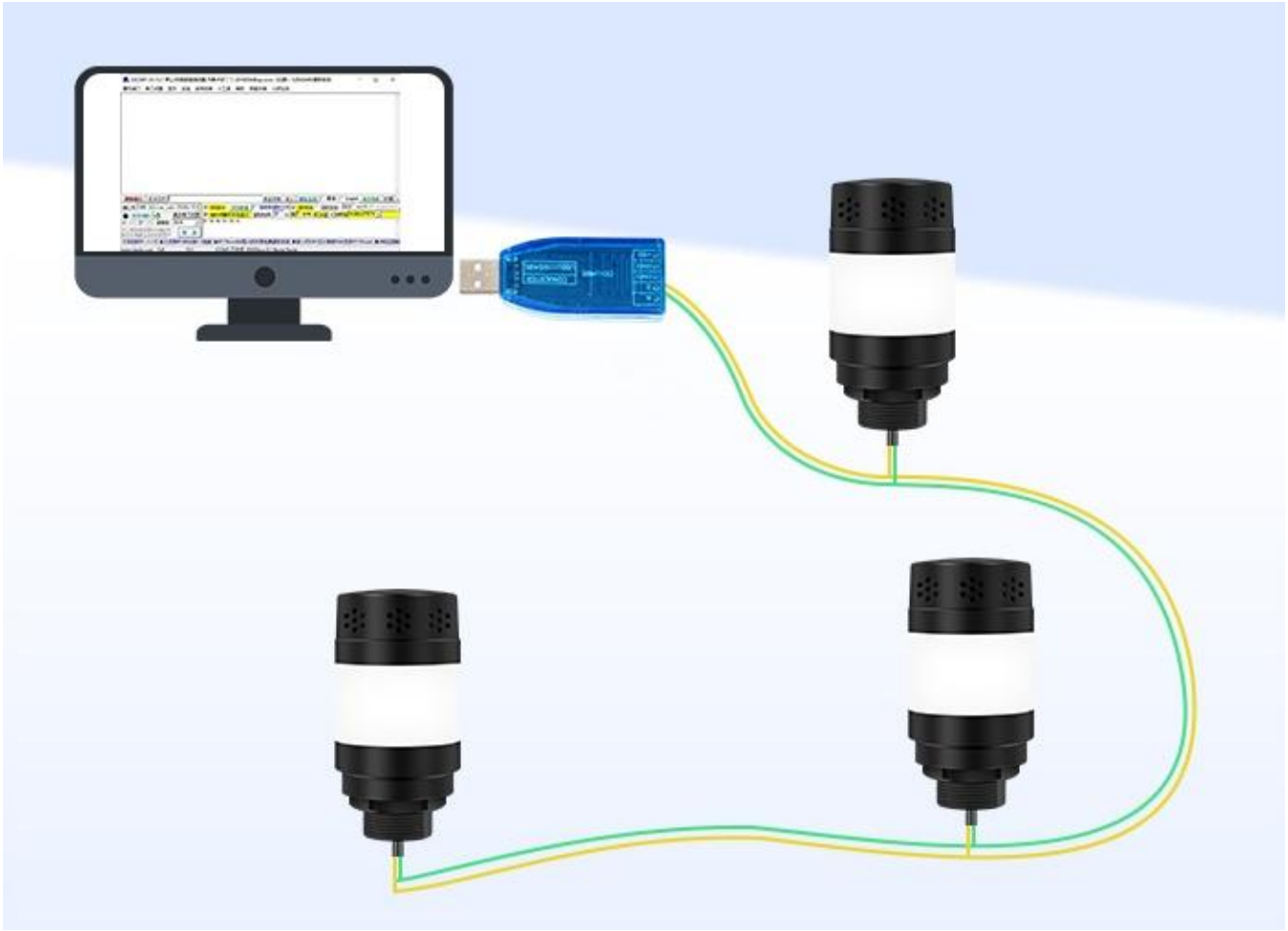
1.3. Dimensions

1.4. Wiring Example



This wiring example shows that the device is connected to a computer via a USB to RS485 converter. Running a serial debugging software on computer, users can debug and test the alarm easily before applying it to a workshop machine or an industrial equipment. When it is connected to a PLC or an industrial controller, in the same way, the yellow wire from the alarm is connected to the RS485 communication port A (+) of the PLC/industrial controller, while the green wire is connected to the RS485 communication port B (-).

If you need to control multiple slave devices, you can refer to the wiring example of RS485 as below.



2. Serial Communication

This product adopts Modbus RTU serial communication protocol based on RS485, and the default baud rate is 9600. Because of its binary representation and compact data structure, Modbus RTU has high communication efficiency and it is widely used in industrial occasions.

Factory default parameters for the device:

Default device address: 01

Default volume level: 30

Default baud rate: 9600

RS485 wiring: Yellow A +/ Green B -

RS485 communication settings:

Baud rate: 9600 (default)

Data bits: 8

Stop bits: 1

Parity: None

Flow control: None

2.1. Communication format

Data bits: 8; Stop bits: 1; Error detection: CRC (Cyclic Redundancy Check)							
Addr	**Fun**	**Data start reg hi**	**Data start reg lo**	**Data #of regs hi**	**Data#of regs lo**	**CRC16_L**	**CRC16_H**
01	06	00	03	00	01	B8	0A
slave address	function code	register address high byte	register address low byte	register data high byte	register data low byte	check low byte	check high byte
Address code (1 byte), function code (1 byte), start address (2 bytes), data (2 bytes), check code (2 bytes)							

Addr: slave address (device address)

Fun: function code

Data start reg hi: data start address - register high byte

Data start reg lo: data start address - register low byte

Data #of reg hi: number of data read - register high byte

Data #of reg lo: number of data read - register low byte

CRC16_H: Cyclic redundancy check high byte

CRC16_L: Cyclic redundancy check low byte

Command format: [address code] + [function code] + [high 8 bits of register address] + [low 8 bits of register address] + [high 8 bits of data] + [low 8 bits of data] + [low 8 bits of check code] +[high 8 bits of check code]

2.1.1. Address code and function code

Address code:

The address field is at the beginning of the frame consisting of one byte, and the hexadecimal number is 0x00-0xFF. The decimal number is 0-255, of which 255 (0xFF) is our super address. These bits identify the user-specified address of the end device that will receive data from the host connected to it. The address of each terminal device must be unique, and only the addressed terminal will respond to queries containing this address. When the terminal sends back a response, the slave address data in the response tells the host which terminal is communicating with.

Function code:

The function field code tells the addressed terminal what function to perform. The following table lists the function codes that we commonly use, as well as their meanings and functions.

Function Code	Name	Function
03	Read holding register	To get the current binary value in one or more holding registers
06	Preset single register	To load a specific binary value into a holding register

03H function code: Read the data of the specified register. Our product uses this function to read the system status, the total number of files and other data, that is, the query function of our product.

06H function code: Write data into the register, that is, write the received data sent by the host into the register set by itself. The application of this function in our product is that after our chip receives the command sent by the host, the chip will store this command into the established register, and then perform corresponding actions on this instruction, that is, the control function of our product.

2.1.2. Control the status of the signal light based on the high byte of the register address

Command: 0x03; Function code: 06H ; Command function: Play the specified track (play the first track in the root directory)								
Format	**Addr**	**Fun**	**Data start reg hi**	**Data start reg lo**	**Data #of regs hi**	**Data#of regs lo**	**CRC_L**	**CRC_H**
Command	01	06	00	03	00	01	B8	0A
	RS485 address	Function code	Register address high byte	Register address low byte	Register data high byte	Register data low byte	Check low byte	Check high byte

(1) The high byte of the register address is used to control the flashing mode and the color of the signal light. At the same time, as soon as the playback is completed, the control state of the signal light returns to the default.

Command format: FF 06 **XY** CMD DH DL CRC_L CRC_H

In the command, XY represent the parameters that control the flashing mode and the color of the signal light (please see the following parameter table of signal light status value). CMD represents the 06H control command (please refer to the serial command list for details). DH and DL represent the value corresponding to the command.

Example: Send FF 06 11 03 00 01 A8 E8 to specify the first track to play and the red light is always on.

(2) Parameter table of signal light status value table (the register address high byte value **XY** corresponds to)

Values that can be used for X	Corresponding function	Values that can be used for Y	Corresponding function
1	Always on	1	Works in red light
2	Slow flash	2	Works in yellow light
3	Quick flash	3	Works in green light
6	Light goes out	0	No light

Note: When XY is 00, the status of the light during playback is the same as the last playback. When the power is turned on again, the default status of the signal light is off.

Reference commands:

Command (xx xx means CRC check)	Note
FF 06 31 0F 01 01 xx xx	Play track 001 in folder 01, and signal light works in red with quick flash
FF 06 23 0F 01 02 xx xx	Play track 002 in folder 01, and signal light works in green with slow flash
FF 06 12 0F 02 01 xx xx	Play track 001 in folder 02, and signal light works in yellow with always on
FF 06 60 0F 02 02 xx xx	Play track 002 in folder 02, and signal light goes out (no light)

2.1.3. Error check field

This field allows the host and terminals to check for errors during transmission. Sometimes due to electrical noise and other interference, a set of data may change through the wire when it is transmitted from one device to another. Error checking can ensure that the host or terminal does not respond to the data that has changed during transmission. This improves the security and efficiency of the system, and the 16-bit cyclic redundancy method (CRC16) is used for error checking.

In the CRC operation, first a 16-bit register is preset to 0FFFFH (each bit is preset to 1), and then the 8 bits in each byte in the data frame are continuously operated with the current value of the register. Only the 8 data bits of each byte are involved in generating the CRC. The start bits, stop bits and the possible use of parity bits do not affect the CRC. When generating the CRC, the 8 bits of each byte are XORed with the contents of the register, and then the result is shifted to the lower bit. the upper bit is supplemented with "0", and the lowest bit (LSB) is shifted out and detected. If it is 1, the register performs an XOR operation with a preset fixed value (0A001H). If the lowest bit is 0, no processing is performed.

The above processing is repeated until the 8 shift operations are performed. When the last bit (the 8th bit) is shifted, the next 8-bit byte is XORed with the current value of the register. Above another 8 shift XOR operations are also performed in the same way. When all the bytes in the data frame are processed, the final value generated is the CRC value.

The above only explains the calculation method and function of the CRC16 cyclic redundancy check. If you don't understand it, you can ignore this part. Generally, we can hand this check code to the software for calculation, and you don't need to calculate it yourself. It doesn't matter if you don't understand it, please refer to the relevant information for details.

2.2. Serial command list

CMD (command)	Function	Command Sent (hexadecimal characters)	Note
	Control Command (06H)		
0x01	Next	FF 06 00 **01** 00 00 xx xx	Play next track
0x02	Previous	FF 06 00 **02** 00 00 xx xx	Play previous track
0x03	Specify playback of a track in the root directory	FF 06 00 **03** 00 01 xx xx	Specify playback of the 1st track
		FF 06 12 **03** 00 02 xx xx	Specify playback of the 2nd track, and signal light works in yellow with always on
		FF 06 31 **03** 00 02 xx xx	Specify playback of the 2nd track, and signal light works in red with quick flash
0x04	Volume up	FF 06 00 **04** 00 00 xx xx	Increase volume
0x05	Volume down	FF 06 00 **05** 00 00 xx xx	Decrease volume
0x06	Specify volume	FF 06 00 **06** 00 1E xx xx	Set volume level to 30 (volume levels from 00 to 30)
0x08	Specify playback of a track in a loop	FF 06 00 **08** 00 01 xx xx	Play the first track in a loop

0x0B	Set baud rate	FF 06 00 **0B** 00 01 xx xx	Set the baud rate to 9600
		FF 06 00 **0B** 00 02 xx xx	Set the baud rate to 19200
		FF 06 00 **0B** 00 03 xx xx	Set the baud rate to 38400
		FF 06 00 **0B** 00 04 xx xx	Set the baud rate to 57600
		FF 06 00 **0B** 00 05 xx xx	Set the baud rate to 115200
		FF 06 00 **0B** 00 06 xx xx	Set the baud rate to 256000
		FF 06 00 **0B** 00 07 xx xx	Set the baud rate to 35250
		FF 06 00 **0B** 00 09 xx xx	Set the baud rate to 4800
0x0C	Reset	FF 06 00 **0C** 00 00 xx xx	Reset the decoder chip
0x0D	Play	FF 06 00 **0D** 00 00 xx xx	/
0x0E	Pause	FF 06 00 **0E** 00 00 xx xx	/
0x0F	Specify playback of a track in a specific folder	FF 06 00 **0F** 02 07 xx xx	Play track 007 in folder 02
		FF 06 00 **0F** 0F FF xx xx	Play track 255 in folder 15
		FF 06 11 **0F** 01 01 xx xx	Play track 001 in folder 01, and signal light works in red with always on
		FF 06 22 **0F** 63 FF xx xx	Play track 255 in folder 99, and signal light works in yellow with slow flash
0x10	Specify playback of a track in a specific folder in a loop	F F 06 00 **10** 01 01 xx xx	Play track 001 in folder 01 in a loop
		F F 06 00 **10** 02 01 xx xx	Play track 001 in folder 02 in a loop
0x11	Play all tracks in a loop	FF 06 00 **11** 00 01 xx xx	Turn it on
		FF 06 00 **11** 00 00 xx xx	Turn it off
0x16	Stop	FF 06 00 **16** 00 01 xx xx	Stop current playback
0x17	Specify loop playback of all tracks in a folder	FF 06 00 **17** 00 01 xx xx	Specify all tracks in folder 01 for loop playback
0x18	Play all tracks in random order	FF 06 00 **18** 00 00 xx xx	All tracks in the storage device are covered
0x19	Set currently playing track for loop playback	FF 06 00 **19** 00 00 xx xx	/
	Stop loop playback of currently playing track	FF 06 00 **19** 00 01 xx xx	/
0xC0	Set the device address	FF 06 00 **C0** 00 01 xx xx	Set the device address to 01
		FF 06 00 **C0** 00 63 xx xx	Set the device address to 99
0xC2	Control visual alarm (signal light) individually	FF 06 00 **C2** 00 11 xx xx	Signal light works in red with always on and does not change the current playback status.
		FF 06 00 **C2** 00 23 xx xx	Signal light works in green with slow flash and does not change the current playback status.
		FF 06 00 **C2** 00 32 xx xx	Signal light works in yellow with quick flash and does not change the current playback status
		FF 06 00 **C2** 00 60 xx xx	Turn off the light
	Set audible alarm only	FF 06 60 03 00 01 xx xx	Play the first track, and the signal light does not light up
Query Command (03H)			
0x42	Query the current playback status	FF 03 00 **42** 00 00 xx xx	See 3.2.1 for details
0x43	Query the current volume	FF 03 00 **43** 00 00 xx xx	See 3.2.2 for details
0x49	Query the number of files in the storage device	FF 03 00 **49** 00 00 xx xx	See 3.2.3 for details
0x4D	Query the currently playing track	FF 03 00 **4D** 00 00 xx xx	See 3.2.4 for details
0x70	Query the current sound and light status	FF 03 00 **70** 00 00 xx xx	See 3.2.5for details

xx xx is the CRC check. FF is the super address, which can be changed to the corresponding device address.

3. Detailed Explanation of Serial Commands

Below we describe some commonly used commands in detail.

The command format is

address code + function code + start address high byte+ start address low byte + data high byte + data low byte + CRC check

FF 06 00 CMD DH DL CRC_L CRC_H

FF is the super address, and the default device address is 01. CMD is the operation code. CRC_L and CRC_H are the cyclic redundancy check.

After sending a control command to the device, it'll return a set of same data immediately.

In the following commands, DH and DL are input values . Users enter the corresponding values according to actual requirements. Note that the command values are all in hex.

3.1. Detailed Explanation of some control commands

3.1.1. Specify playback of a track in the root directory (0x03)

This command is used to specify playback of a track in the root directory in physical index order. The selection range of tracks is 1-3000. In fact, more tracks can be supported. Because of file management reasons, supporting too many tracks will cause the system to operate slowly. Generally, most of applications do not need to support so many files.

1). If you choose the 100th track, firstly convert 100 to hexadecimal, the default is double byte, which is 0x0064 (DH=0x00, DL=0x64).
2). Other operations can be deduced and so on.
Command format: FF 06 00 03 DH DL CRC_L CRC_H
Example: Send the command “01 06 00 03 00 01 B8 0A” or "FF 06 00 03 00 01 AD D4" to specify playback of the first track in the root directory (DH = 0x00, DL=0x01)
Reference Commands:

Command (xx xx represent CRC)	Function
FF 06 00 **03** 00 01 xx xx	Specify playback of the 1st track
FF 06 00 **03** 00 63 xx xx	Specify playback of the 99th track
FF 06 00 **03** 00 FF xx xx	Specify playback of the 255th track
FF 06 00 **03** 0B B8 xx xx	Specify playback of the 3000th track

3.1.2. Specify volume (0x06)

1). There are 31 volume levels (00-30). If you want to set the volume, just send the corresponding command directly.
2). Sending a command to set the volume will be reset when the device restarts. If you need the power-off memory, you can set the volume through a configuration file. Please refer to 3.1.8 about the descriptions for the configuration file.

Command format: FF 06 00 06 00 DL CRC_L CRC_H (DL is the parameter of the volume to be set).
Example: Send the command "01 06 00 06 00 1E E9 C3" or “FF 06 00 06 00 1E FC 1D” to set the volume level to 30. 30 is 0x1E in hex, so DL=0x1E.

Reference Commands:

Command (xx xx represent CRC)	Function
FF 06 00 **06** 00 00 xx xx	Set the volume level to 00 (mute)
FF 06 00 **06** 00 19 xx xx	Set the volume level to 25

3.1.3. Set baud rate (0x0B)

1). The default baud rate is 9600. If you need to change the baud rate, you can use the command 0x0B to modify it.

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). After setting the baud rate, the device will remember it, and when the device is restarted, the baud rate will become the set one.

Command format : FF 06 00 **0B** 00 DL CRC_L CRC_H (DL is the parameter of the baud rate to be set. See the table below for details)

DL parameters	Corresponding baud rate	DL parameters	Corresponding baud rate
0x01	9600	0x05	115200
0x02	19200	0x06	256000
0x03	38400	0x07	35250
0x04	57600	0x09	4800

Example: Send the command “01 06 00 0B 00 02 79 C9” or “FF 06 00 0B 00 02 6C 17” to set the baud rate to 19200 (DL=0x02)

Reference commands:

Command (xx xx represent CRC)	Function
FF 06 00 **0B** 00 01 xx xx	Set the baud rate to 9600
FF 06 00 **0B** 00 05 xx xx	Set the baud rate to 115200

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

Command format: FF 06 00 0F DH DL CRC_L CRC_H

DH represents the name of the folder, and supports up to 99 folders that is 01-99 (DH=0x01-0x63)

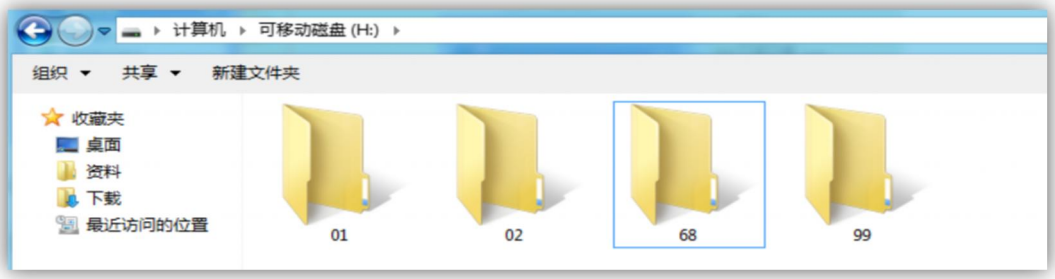
DL represents the track, and supports up to 255 tracks that is 001-255 (DL=0x01-0xFF)

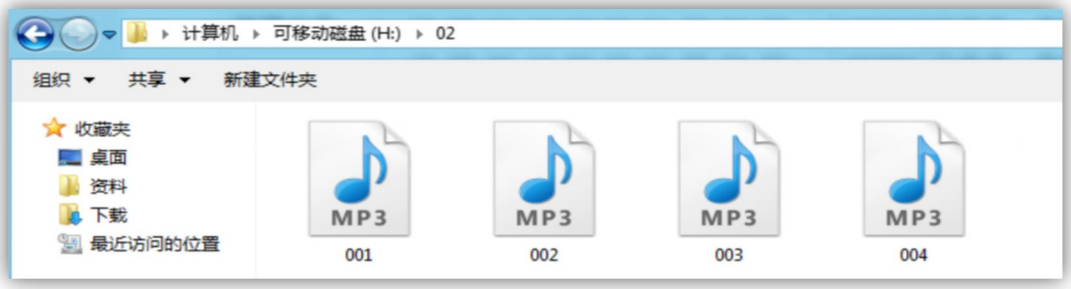
Example: Send the command “01 06 00 0F 02 01 79 69” or “FF 06 00 0F 02 01 6C B7” to play track “001xxx.mp3” in folder “02” (DH=0x01, DL=0x01)

Reference Commands:

Command (xx xx represent CRC)	Function
FF 06 00 **0F** 01 01 xx xx	Specify playback of track 001.mp3 in folder 01
FF 06 00 **0F** 01 63 xx xx	Specify playback of track 099.mp3 in folder 01
FF 06 00 **0F** 63 FF xx xx	Specify playback of track 255.mp3 in folder 99

The following two screenshots illustrate the designation of folder names and file names. Please name the folders and files in strict accordance with our instructions.





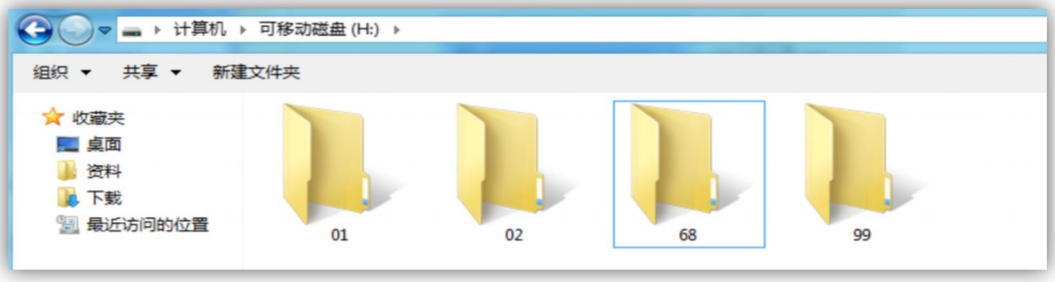
3.1.5. Specify playback of a track in a specific folder in a loop (0x10)

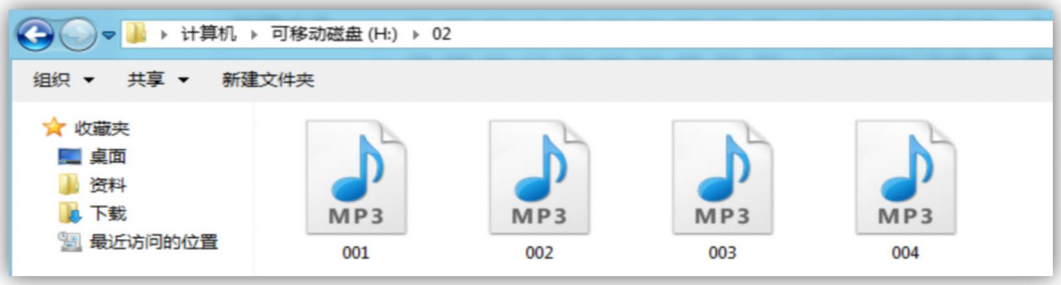
Command format: FF 06 00 10 DH DL CRC_L CRC_H

DH represents the name of the folder, and supports up to 99 folders that is 01-99 (DH=0x01-0x63)
DL represents the track, and supports up to 255 tracks that is 001-255 (DL=0x01-0xFF)
Example: Send the command “01 06 00 10 02 01 48 AF” or “FF 06 00 10 02 01 5D 71” to specify the track 001xxx.mp3 in the folder 02 to play in a loop.
Reference commands:

Command (xx xx represent CRC)	Function
FF 06 00 **10** 01 01 xx xx	Play track 001xxx .mp3 in folder 01 in a loop
FF 06 00 **10** 01 63 xx xx	Play track 099xxx .mp3 in folder 01 in a loop
FF 06 00 **10** 63 FF xx xx	Play track 255xxx .mp3 in folder 99 in a loop

The following two screenshots illustrate the designation of folder names and file names. Please name the folders and files in strict accordance with our instructions.





3.1.6. Specify loop playback of all tracks in a folder (0x17)

Command format: FF 06 00 17 00 DL CRC_L CRC_H (DL represents the name of the folder and supports up to 99 folders that is 01-99, DL=0x01-0x63)

Example: Send the command “01 06 00 17 00 01 F8 0E” or “FF 06 00 17 00 01 ED D0” to play all tracks in a loop in the folder 01 (DL=0x01)

Notes:

1). The naming method of the folder must be "01" to "99", and cannot exceed 99.

2). Once the loop playback starts, you can use play/pause/previous/next. None of these operation commands will interrupt the current loop playback state. That is to say, after sending the “next” command, the current folder will still be in loop playback mode.

3). Users can send a stop command to end the loop playback and return to the trigger playback state.

4). Each time a track is played, there will be a frame of returned data, corresponding to the track that has just been played.

Reference Commands:

Command (xx xx represent CRC)	Function
FF 06 00 **17** 00 01 xx xx	Specify loop playback of tracks in folder 01
FF 06 00 **17** 00 63 xx xx	Specify loop playback of tracks in folder 99

3.1.7. Set currently playing track for loop playback (0x19)

Command format: FF 06 00 **19** 00 DL CRC_L CRC_H (DL= 0x00, to start loop playback; DL=0x01, to end loop playback)

Example: Send the command “FF 06 00 19 00 00 4D D3” (DL= 0 x00) to start the loop playback.
Send the command “FF 06 00 19 00 01 8C 13” (DL= 0x01) to end the loop playback.

1). If you send this command during playback, the current track will be played in a loop. If it is currently in pause or stop state, the device will not respond to this command.

2). If you want to stop the loop playback, you can send the end command, which will stop the current track after playing.

3.1.8. Set the device address (0xC0)

Command format: FF 06 00 C0 00 DL CRC_L CRC_H

DL represents the device address that needs to be set. It can be set to 1-254 (0x01-0xFE in hex).

Example: Send the command “01 06 00 C0 00 02 08 37” or “FF 06 00 C0 00 02 1D E9” to set the device address to 02.

Reference commands:

Command (xx xx represent CRC)	Function
FF 06 00 **C0** 00 01 xx xx	Set the device address to 01
FF 06 00 **C0** 00 63 xx xx	Set the device address to 99
FF 06 00 **C0** 00 F7 xx xx	Set the device address to 247

1). After sending the command, the device address will take effect immediately and can be remembered when the power is turned off.

2). The address setting range is 1-254, so the value range is 0x01-0xFE for the DL.

3). The function of setting the address is that multiple devices can be connected to the RS485 bus, which is equivalent to giving each device a unique name, so that each device can be controlled individually. For details, you can search for the principle of RS485.

4). This product also supports setting the device address through a configuration file (text file), and the configuration file takes priority. For details, please refer to the details as below.

As you can see there are 5 digits in the configuration file (it's a text file). The first digit "0" represents the trigger mode in key control mode, the second and third digits "10" represent the volume setting (00-30 can be set here), and the fourth and fifth digits "02" are the device address (01-99 can be set here).

Notes:

a). When you create a configuration file like this, you don't need to care the first digit, because it's purposed for key control mode, but it must exist.

b). This configuration file must be placed on the root directory of the memory.

3.1.9. Control visual alarm (signal light) individually (0xC2)

Command format: FF 06 00 C2 00 XY CRC_L CRC_H

X represents the output mode and Y represents the color of the light.

Values that can be used for X	Corresponding function	Values that can be used for Y	Corresponding function
1	Always on	1	Works in red light
2	Slow flash	2	Works in yellow light
3	Quick flash	3	Works in green light
		0	No light

1). This command (0xC2) is to independently control the working status of the signal light and does not affect the current audio playback status.

2) After the device receives the related command, the signal light will always light up according to the command. If you need to turn off the signal light, you need to send an command to turn it off or send other commands to change the working status of the signal light.

3) When the value of Y is 0, the signal light is going to be turned off. For example, sending the command "FF 06 00 C2 00 60 3C 24" to turn off of the signal light without affecting the current audio playback.

Reference commands:

Command (xx xx represent CRC)	Function
FF 06 00 **C2** 00 11 xx xx	Signal light works in red and always on
FF 06 00 **C2** 00 12 xx xx	Signal light works in yellow and always on
FF 06 00 **C2** 00 13 xx xx	Signal light works in green and always on
FF 06 00 **C2** 00 21 xx xx	Signal light works in red and slow flash
FF 06 00 **C2** 00 22 xx xx	Signal light works in yellow and slow flash
FF 06 00 **C2** 00 23 xx xx	Signal light works in green and slow flash
FF 06 00 **C2** 00 31 xx xx	Signal light works in red and quick flash
FF 06 00 **C2** 00 32 xx xx	Signal light works in yellow and quick flash
FF 06 00 **C2** 00 33 xx xx	Signal light works in green and quick flash
FF 06 00 **C2** 00 60 xx xx	Signal light is off

3.1.10. Set audible alarm only (without turning on the signal light)

Command format: FF 06 60 CMD DH DL CRC_L CRC_H

(60 represents turning off the signal light, and please refer to 3.1.2 for details; CMD represents the operation code, and please refer to serial command list for details)

Example: Send the command “FF 06 60 0F 01 01 72 47” to play track “001xxx.mp3” in folder “01” and the signal light will not light up. We set the current alarm status through the "register address high byte". If the value is 60, it means that the current playback does not turn on the light, but audio playback only. The other control commands are the same.

Reference commands:

Command (xx xx represent CRC)	Function
FF 06 60 **03** 00 02 xx xx	Play the 2nd track in the root directory, and the signal light is currently off
FF 06 60 **0F** 01 02 xx xx	Play track 002.mp3 in folder 01, and the signal light is currently off
FF 06 60 **18** 00 00 xx xx	Play in random, and the signal light is currently off
FF 06 60 **19** 00 00 xx xx	Set currently playing track to loop playback, and the signal light is off.

3.2 . Detailed explanation of some query commands

3.2.1. Query the current playback status (0x42)

Send the command “FF 03 00 42 00 00 F0 00” to query the current playback status of the device (CMD=0x42)

Returned data: “01 03 02 04 01 7B 44” (DH=0x04, DL=0x01, indicating the device is playing)

“01 03 02 04 02 3B 45” (DH=0x04, DL=0x02, indicating the playback is paused)

“01 03 02 04 00 BA 84” (DH=0x04, DL=0x00, indicating the device stops playing)

DL Value from Returned Data	Corresponding Status
0x00	Playback is stopped

0x01	Playing
0x02	Playback is paused

Note: The address code 0x01 in the returned data indicates that the current device address is 01.

3.2.2. Query the current volume (0x43)

Send the command “FF 03 00 43 00 00 A1 C0” to query current volume (CMD=0x43)
Returned data: “01 03 02 A1 1E 41 DC” (the current volume level is 30, DL=0x1E)
The DL value in the returned data represents the volume value. Converting its hexadecimal number to a decimal number is the current volume level. For example, 0x1E converted to decimal is 30, which means the current volume level is 30. The address code 0x01 in the returned data indicates that the current device address is 01.

3.2.3. Query the number of files in the storage device (0x49)

Send the command “FF 03 00 49 00 00 81 C2” to query the total number of files in the storage device.
Returned data: “01 03 02 00 08 B9 82” (the total number of files is 8).
The DH and DL bytes in the returned data represent the total number of files in the device. For example, converting DH=0x00, DL=0x08 to decimal is 08, which means that the total number of files is 8. Generally, we do not save so many files, so just look at the DL value. The address code 0x01 in the returned data indicates that the current device address is 01.

3.2.4. Query the currently playing track (0x4D)

Send the command “FF 03 00 4D 00 00 C0 03” to query the currently playing track.

Return data: “01 03 02 00 02 39 85” (currently the second track in physical order is playing or just finished playing)

The DH and DL value in the returned data represent the track currently playing or just finished. For example, converting DH=0x00, DL=0x02 to decimal is 02, which indicates that the current track is the second one in the physical order. Generally, we don’t save so many files, so just look at the DL value. The address code 0x01 in the returned data indicates that the current device address is 01.

3.2.5. Query the current sound and light status (0x70)

Send the command “FF 03 00 70 00 00 51 CF” to query the current sound and light status.
Return data: “01 03 02 DH DL CRC_L CRC_H”

The returned data includes information such as the device address, current playback status, and current status of signal light and it’s

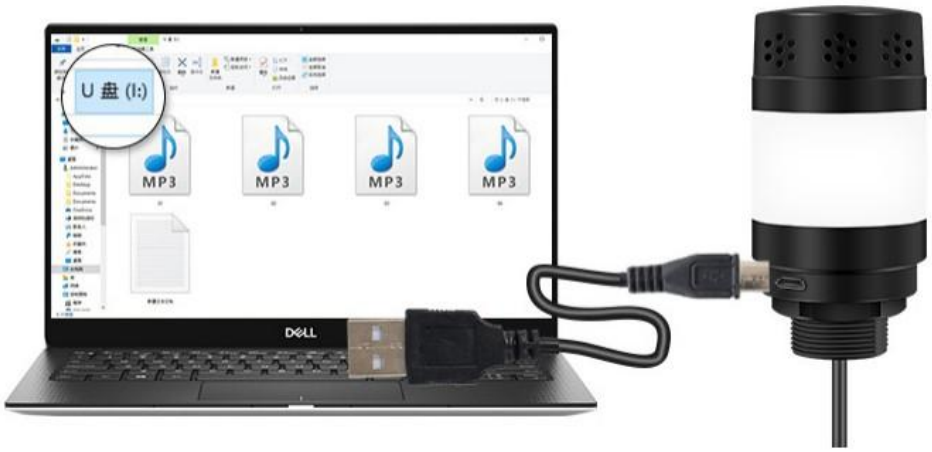
color.

Let's take the returned data “01 03 02 00 12 38 49” as an example. The current device address is 01. DH=0 x00 , which means that no sound is currently playing. DL=0X12, which means that the signal light lights up yellow and it’s always on.

In the returned data, the address code indicates the current device address, DH value indicates the audio playback status. For example, DH=0x01 indicates that the first track in the physical order is currently playing, and the DH value range is 0x00-0xFF, that is, the maximum supported query range is tracks 001-255. When DH=0x00, it means that currently no sound is playing; DL represents the light status. For example, DL=0x11 means that the current light status is red and always on. The corresponding status of DL value is shown in the table as below.

DL value	Corresponding light status	DL value	Corresponding light status
11	Red and always on	23	Green and slow flash
12	Yellow and always on	31	Red and quick flash
13	Green and always on	32	Yellow and quick flash
21	Red and slow flash	33	Green and quick flash
22	Yellow and slow flash	10 / 20 / 30 / 60 /06... etc .	Signal light goes out
If the returned value is not the values listed above, the light status should be off. If the command sent does not set the light status, the returned data may not correspond to the actual light status. For setting the light status, please refer to 3.1.2.			

4. How to Replace Audio Files



As you can see, this product has a micro USB port. It can be connected to computer through a USB data cable to replace audio files.The internal flash memory of the device will be detected as a USB flash drive on computer. You can directly put the files in the root directory of the memory, or create folders first like 01, 02...., and put the files in these folders. Files in folders need to be named as 001.mp3, 002.mp3.....

Notes

1. Before replacing your own audio files, it is better to format the USB drive first and then copy your own files into it to avoid playback errors due to hidden files that probably exist.
2. The audio files must be in MP3 format and must have a sampling rate supported by the product. This product supports sampling rate (KHz): 8/11.025/12/16/22.05/24/32/44.1/48.
3. When connecting to computer to replace audio files, please disconnect the power supply first. After replacing the audio files, you need to disconnect the device from computer and then power on the device.