

# User Manual

(TD5580A chip)

File Status:	File Identification	Restricted
draft[ ]	author	Mark
Officially released[ÿ]	Current version	V1.3
Modifying [ ]	Completion Date	2024.04.19

## Table of contents

Introduction.....	1
1. DEMO module introduction.....	1
2. Chip Features.....	2
3. Application scenarios.....	2
4. Hardware Description.....	3
4.1 Pin Introduction.....	3
4.2 Technical Specifications.....	4
4.3 Electrical parameters.....	4
5. Serial Port Control.....	5
5.1 Serial communication protocol.....	5
5.2 Serial communication instructions.....	5
5.3 Data returned when the module is powered on.....	7
5.3.1 Data returned after a track is played.....	8
5.3.2 Data returned by the module response.....	8
5.3.3 Data returned by module error.....	9
5.3.4 Device plugging and unplugging messages.....	9
5.4 Detailed explanation of serial port commands.....	9
5.4.1 Playing a Specified Track.....	10
5.4.2 Specifying the playback volume.....	10
5.4.3 Specifying playback devices.....	10
5.4.4 Playing in a specified folder.....	11
5.4.5 All loop playback instructions.....	12
5.4.6 Inserting advertisements in the ADVERT folder.....	12
5.4.7 A single folder supports 1000 tracks.....	13
5.4.8 All loop playback instructions.....	13
5.4.9 Single song loop play command.....	13
5.4.10 Single song loop play command.....	14
5.4.11 Playback Stop Command.....	14

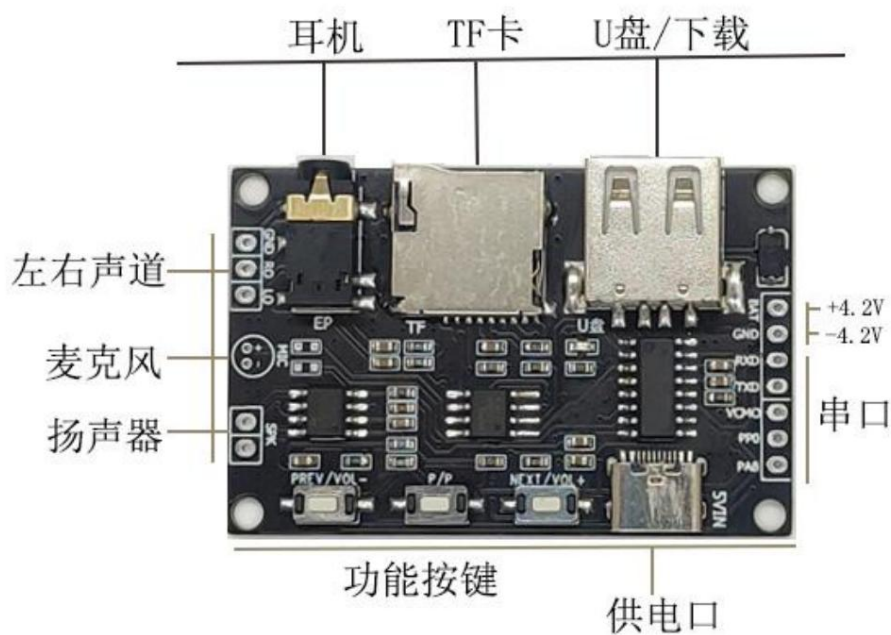
5.4.12 Loop playback of specified folders.....	14
5.4.13 Randomly play device files.....	14
5.4.14 Set the current track to loop playback.....	14
5.4.15 Combined broadcast.....	15
5.4.16 Turning DAC on and off.....	15
6. ADKEY Key Description.....	15
7. Customized functions can be developed.....	15
8. FAQ:.....	16

## Introduction

TD5580A is a powerful high-quality voice decoding chip. It supports MP3, WAV and other decoding formats, SPI-Flash, TF card, USB disk and other storage media. The software also supports industrial-grade serial communication protocol. Users only need to use simple serial port commands to easily complete the specified voice playback. In addition, the chip also has advanced playback functions such as insert, file index playback, and combined broadcast. It is easy to use, stable and reliable.

## 1. DEMO module introduction

The TD631 module is a DEMO board for the TD5580A chip, which is mainly used to facilitate customers to quickly verify and debug functions. The module automatically starts up when powered on by default, and enters the waiting command control state after startup. Users can control audio playback through commands and buttons.



## 2. Chip characteristics

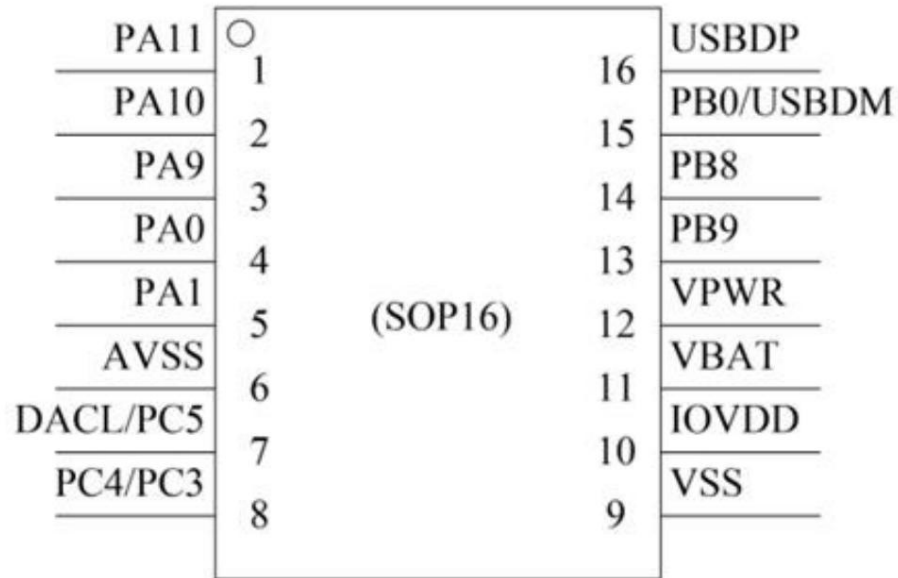
- Support MP3 and WAV decoding formats;
- Support 6 types of sound effects including pop, jazz, classic, etc.;
- Supports FAT16/FAT32/exFAT file systems;
- Supports up to 64G TF card and USB flash drive, 16M bytes of NOR FLASH;
- Support sampling rates (KHz): 8/11.025/12/16/22.05/24/32/44.1/48/64/88.2/96;
- 24-bit DAC output, signal-to-noise ratio supports 93dB;
- Multiple control modes available: IO control mode, serial port mode, AD button control mode;
- USB interface to update voice files, no need to install any software;
- Operating temperature: -40° to +85°, can be used in industrial scenarios such as industrial control;

## 3. Application scenarios

- In-car navigation voice broadcast;
- Highway transport inspection and toll booth voice prompts;
- Voice prompts for security checks at railway stations and bus stations;
- Voice prompts in power, communications, and financial business halls;
- Voice prompts for vehicle entry and exit channel verification;
- Voice prompts for public security border inspection channels;
- Voice announcements for safe driving of electric sightseeing vehicles;
- Automatic alarm for mechanical and electrical equipment failure;
- Fire voice alarm prompts;
- Other TTS high-quality announcements;

## 4. Hardware Description

### 4.1 Pin Introduction



Pin No.	Pin Name	Functional Description	Remark
1	PA11	I/O	SDCMD
2	PA10	I/O	SDDAT
3	PA9	I/O	SDCLK
4	PA0	I/O	MUTE (low play, high mute)
5	PA1	I/O	MIC Input Channel (reserved)
6	ASSS	GROUND	Ground
7	DACL	OUTPUT	DAC Channel
8	PC4	I/O	AD-KEY
9	GND	GROUND	Ground
10	IOVDD	POWER	Output voltage 3.3V, current 100MA
11	VBAT	POWER	Battery Power Supply
12	VPWR/P00	I/O	BLED (high level lights up)
13	PB9	I/O	TXD
14	PB8	I/O	RXD
15	USBDM	I/O	USBDM
16	USBDP	I/O	USBDP

## 4.2 Technical specifications

name	Function
Audio Format	Supports MP3 audio files (FLASH) with sampling rate of 8~48KHz and bit rate of 8~128Kbps Support MP3 audio files with sampling rate of 8~48KHz and bit rate of 8~320Kbps (TF card and U disk) Supports WAV audio files with sampling rates from 8 to 44.1KHz.
Memory capacity	Support 4Mbit~128MbitSPI-FLASH, TF card supports up to 64GB, USB flash drive supports up to 64GB Supported file systems: FAT16, FAT32, exFAT (NTFS is not supported)
USB interface	Full Speed 2.0
Chip power supply voltage	DC2.7~5V
IO port level	3.3V TTL level
Working temperature	-40~85 degrees
humidity	5%~95%

## 4.3 Electrical parameters

name	mark	Conditions	Min	Typ	Max	Unit
Input voltage: The chip works normally with an input voltage of 2.2V to 5V				2.2	3.7	5 V
Quiescent current	No load current No load -				10	mA
Working current	Current in playback state	4R/3W 1K signal Maximum average current			750	mA
Soft shutdown current	The current in the soft shutdown state -			1.5	2	2.5 uA
SNR	Signal-to-Noise				93	dB
THD+N	Ratio Total Harmonic Distortion				-80	dB
DAC OUTPUT DAC	maximum output amplitude				750	mVrms
FrequencyRes	Frequency Response Curve			20		20K HZ

## 5. Serial port control

### 5.1 Serial communication protocol

Fixed baud rate: 9600 bps Data bits: 8 Stop bits: 1 Parity bit: none Flow control: none			
Instruction format	HEAD	ADDR	LEN CMD ACK DATA CHECKSUM END
HEAD	The starting byte is fixed at 0x7E		
ADDR	Address 1 byte fixed 0xFF		
JUST	Length 1 byte excluding frame start, frame end, checksum and length		
CMD	Command 1 byte operation, query		
ACK	Feedback 1 Byte 1-feedback is required after the command is issued; 0-no feedback		
DATA	Parameter n bytes		
CHECKSUM	Checksum 2 bytes CHECKSUM = 0-(ADDR + LEN + CMD + ACK + DATA)		
END	End 1 byte fixed 0xEE		
Precautions	<p>Voice module serial port working voltage 3.3V</p> <p>All commands are in hexadecimal format, and the interval between each command is no less than 300ms.</p> <p>There are two types of check code: with or without check code, which can be selected according to the situation</p>		

### 5.2 Serial communication instructions

#### Control instructions

Order	Function	parameter	instruction
0x01	Next song		7E FF 06 01 00 00 00 EF
0x02	Previous song		7E FF 06 02 00 00 00 EF
0x03	Specify the track number (NUM) to play 0-2999 in the physical order of file storage		7E FF 06 03 00 00 00 EF
0x04	Volume+		7E FF 06 04 00 00 00 EF
0x05	volume-		7E FF 06 05 00 00 00 EF
0x06	Specify volume	0-30	7E FF 06 06 00 00 XX EF
0x07	Specify EQ (0/1/2/3/4/5)	0-Normal; 1-Rock; 3-Pop; 4-Classic; 5-Country; 6-Jazz;	7E FF 06 07 00 00 XX EF
0x08	Specified track single loop 0-2999		7E FF 06 08 00 XX XX EF
0x09	Specify playback device	1-U disk; 2-SD card; 3-AUX (not supported); 4-FLASH; 5-PC; 6-SLEEP;	7E FF 06 09 00 00 XX EF
0x0A	Enter low power consumption	(Not supported)	7E FF 06 0A 00 00 00 EF



0x0B Reserved			7E FF 06 0B 00 00 00 EF
0x0C Chip reset			7E FF 06 0C 00 00 00 EF
0x0D Play			7E FF 06 0D 00 00 00 EF
0x0E Pause			7E FF 06 0E 00 00 00 EF
0x0F	Specify folder and file number Play	The folder name is a two-digit number "01", (1-99) Example: 01_Rock The file name is three digits "001", (1-255) Example:001_fireworks.mp3	7E FF 06 0F 00 XX XX EF
0x10 Amplification settings (none)		(Not supported yet)	7E FF 06 10 00 00 00 EF
0x11 Play all loops		1- Play all tracks in a loop (by physical storage Sequence); 0-stop loop playback	7E FF 06 11 00 00 XX EF
0x12	Specify the MP3 folder Playing tracks in	1-9999, the folder name must be "MP3" 4-digit number of the files in the folder 0001	7E FF 06 12 00 XX XX EF
0x13	Insert ADVERT file Advertisements under the clip	1-9999, the folder name must be "ADVERT" The first 4 digits of the files in the folder are numbered 0001	7E FF 06 13 00 XX XX EF
0x14	Specify folder and file number Play	The high four bits of DH are used as the folder number (1-15) The lower four digits and DL are the file number (1-2999)	7E FF 06 14 00 XX XX EF
0x15 Stop inserting ads			7E FF 06 15 00 00 00 EF
0x16 Stop playing		Stop all playback tasks	7E FF 06 16 00 00 00 EF
0x17 Loop playback of specified folder [DH=0] [DL: folder number 0-99]			7E FF 06 17 00 00 00 EF
0x18 All files are played randomly in physical order			7E FF 06 18 00 00 00 EF
0x19 Current track loop play 0-Open single loop; 1-Close single loop			7E FF 06 19 00 00 00 EF
0x1A Enable DAC high impedance 0- Turn off DAC high; 1- Enable DAC high impedance			7E FF 06 1A 00 00 00 EF
0x1B Power-on volume memory		0-Power-on memory; 1-30 Fixed power-on volume	7E FF 06 1B 00 00 00 EF
0x1C Set baud rate		0-9600;1-19200;2-38400; 3-57600; 4-115200; Restart to take effect	7E FF 06 1C 00 00 00 EF
0x1D Restore factory settings		Data restored to initial value	7E FF 06 1D 00 00 00 EF
0x20 Combination Report			7E FF XX 20 00 XX XX IF

## Query command

Order	Function	parameter	instruction
0x3A Device insertion		1. Insert UDISK; 2. Insert TF card; 3. PC	7E FF 06 3A 00 00 00 EF
0x3B Device unplugged		insert 1. Pull out UDISK; 2. Pull out TF card; 3. PC	7E FF 06 3B 00 00 00 EF
0x3C	UDISK Playback Completed	Pull out	7E FF 06 3C 00 00 00 EF
0x3D	TF card playback completed		7E FF 06 3D 00 00 00 EF

0x3E	Flash playback completed		7E FF 06 3E 00 00 00 EF
0x3F	Send initialization parameters (set Online status)	1.U disk; 2.TF card; 3.PC; 4.FLASH	7E FF 06 3F 00 00 00 EF
0x40	Return error, request resend	1. The file system is initializing; 2. Sleep mode only supports specified devices; 3. One frame of data has not been received completely; 4. Checksum error; 5. The file designation exceeds the range; 6. The specified file was not found; 7. The current state does not accept interruptions;	7E FF 06 40 00 00 00 EF
0x41	Response		7E FF 06 41 00 00 00 EF
0x42	Query current status	0. Completed; 1. Currently playing; 2. Played 3. No device is online/sleeping;	7E FF 06 42 00 00 00 EF
0x43	Query the current volume	Return 0-30;	7E FF 06 43 00 00 00 EF
0x44	Query the current EQ		7E FF 06 44 00 00 00 EF
0x45	Query the current playback mode		7E FF 06 45 00 00 00 EF
0x46	Query the current software version		7E FF 06 46 00 00 00 EF
0x47	Query the total number of files on the USB disk		7E FF 06 47 00 00 00 EF
0x48	Query the total number of files in the TF card		7E FF 06 48 00 00 00 EF
0x49	Query the number of FLASH files		7E FF 06 49 00 00 00 EF
0x4A	Reserved		7E FF 06 4A 00 00 00 EF
0x4B	Query the current track of the USB disk		7E FF 06 4B 00 00 00 EF
0x4C	Query the current track of TF card		7E FF 06 4C 00 00 00 EF
0x4D	Check FLASH current track		7E FF 06 4D 00 00 00 EF

### 5.3 Data returned when the module is powered on

USB flash drive -- Online	7E FF 06 3F 00 00 01 xx xx EF	The relationship between the devices is OR
TF Card--Online	7E FF 06 3F 00 00 02 xx xx EF	
PC -- Online	7E FF 06 3F 00 00 04 xx xx EF	
FLASH -- Online	7E FF 06 3F 00 00 08 xx xx EF	
U disk, TF card--Online	7E FF 06 3F 00 00 03 xx xx EF	

(1) When the module is powered on, it takes a certain amount of time to initialize. This time depends on the USB flash drive, TF card, Flash, etc. The time is determined by the number of files in the device, usually 1.5~3S. If this time is exceeded, the module will be initialized. The data has not been sent out, indicating that the module initialization error, please reset the power of the module and check the hardware connection.

(2) Module initialization data includes online devices, for example, sending 7E FF 06 3F 00 00 01 xx xx EF.

(3) DL=0x01 means that during the power-on process, only the USB flash drive is online. For other data, please refer to the table below. The relationship between the devices is OR.

(4) The MCU must wait for the module initialization instruction to be issued before sending the corresponding control instruction. Otherwise, the module will not process the sent instruction, which will also affect the normal initialization of the module.

### 5.3.1 Data returned after the track is played

The USB drive has finished playing the first song	7E FF 06 3C 00 00 01 xx xx EF U disk playback of the first track is completed
The USB drive finishes playing the second track	7E FF 06 3C 00 00 02 xx xx EF U disk playback of the second track is completed
TF card finishes playing the first song	7E FF 06 3D 00 00 01 xx xx EF TF card playback of the first track completed
TF card finishes playing the second song	7E FF 06 3D 00 00 02 xx xx EF TF card playback of the second track is completed
FLASH finished playing the first song	7E FF 06 3E 00 00 01 xx xx EF FLASH Playing the first track is finished
FLASH finished playing the second song	7E FF 06 3E 00 00 02 xx xx EF FLASH Playing the second track is finished

(1) In response to many trigger-type playback requirements, our module has been modified to automatically enter the stop state after playing a song. If the user needs such an application, he only needs to specify the song to play. In this way, the module will automatically stop after the song is played and wait for instructions.

(2) In addition, we specially open up an IO as a status indicator for decoding and stopping. In the playback state, it outputs a low level (many amplifiers have a mute pin, which can be directly controlled through this IO); in the playback pause state, it outputs a high level, and in the module sleep state, it also outputs a high level.

(3) For continuous playback applications, this can be achieved as follows: if the USB flash drive finishes playing the first song, it will return 7E FF 06 3C 00 00 01 xx xx EF. 3C—indicates the USB flash drive command. 00 01—indicates the song that has been played. At this time, send the command to play the next song, and the songs can be played in sequence.

(4) After the module is powered on and initialized normally, the module will automatically enter the device playback state and stop decoding, waiting for the user to send relevant playback instructions.

(5) In addition, after specifying the device, the user needs to wait for 200ms before sending the specified track. This is because once the track is specified, the system will initialize the file system of the specified device. If the specified track command is sent immediately, the module will not be able to receive it.

### 5.3.2 Data returned by the module response

FLASH finished playing the first song	7E FF 06 3E 00 00 01 xx xx EF FLASH Playing the first track is finished
---------------------------------------	---

(1) In order to enhance the stability of data communication, we have added response processing. The ACK byte is used to set whether a response is required. The advantage of this is that it ensures that there is a handshake signal for each communication. Receiving a response means that the module has successfully received the data sent by the MCU and will process it immediately.

(2) For general applications, customers can freely choose not to add this response processing.

### 5.3.3 Data returned by module error

Back to Busy	7E FF 06 40 00 00 01 xx xx EF module during file system initialization	
Currently in sleep mode	7E FF 06 40 00 00 02 xx xx EF Sleep mode only supports specified devices	
Serial port receiving error	7E FF 06 40 00 00 03 xx xx EF The serial port has not received a frame of data.	
Verification error	7E FF 06 40 00 00 04 xx xx EF and checksum error	
The specified file is out of range	7E FF 06 40 00 00 05 xx xx The specified EF file exceeds the setting range.	
The specified file was not found	7E FF 06 40 00 00 06 xx xx EF The specified file was not found.	
Insert command error	7E FF 06 40 00 00 07 xx xx EF The current state does not accept interruption	

(1) In order to enhance the stability of data communication, we have added a data error handling mechanism.

All data in different formats will have information feedback.

(2) In harsh environments, it is strongly recommended that customers process this command. If the application environment is normal, it is not necessary.

Use processing.

(3) The module returns busy. Basically, it will return when the module is powered on and initialized, because the module needs to initialize the file system.

(4) After the module is powered on, it enters the device state. The order of devices is U disk -- TF card -- FLASH.

If both the hard disk and TF card are offline, the module will automatically enter the FLASH state.

Enter sleep state.

### 5.3.4 Device plug-in and unplug-out messages

USB flash drive inserted	7E FF 06 3A 00 00 01 xx xx EF	
TF Insert	7E FF 06 3A 00 00 02 xx xx EF	
PC Insertion	7E FF 06 3A 00 00 04 xx xx EF	
Remove the USB flash drive	7E FF 06 3B 00 00 01 xx xx EF	
TF pull out	7E FF 06 3B 00 00 02 xx xx EF	
PC unplug	7E FF 06 3B 00 00 04 xx xx EF	

(1) In order to enhance the flexibility of the module, we have added command feedback for device insertion and removal to facilitate users to know the working status of the channel module.

(2) When the device is inserted, we enter the device waiting state by default. If the user inserts a USB flash drive with a light, you can see the USB flash drive light flashing, and you can also receive serial port messages from the device being plugged in.

### 5.4 Detailed explanation of serial port commands

The following is a detailed description of the key points of the serial port instructions:

### 5.4.1 Playing a specific track

The instructions we give support the playback of specified tracks. The song selection range is 0~2999. In fact, it can support more. Because of the file system, supporting too many songs will cause the system to operate slowly. General applications do not need to support so many files.

(1) For example, to select the first song to play, the sending part of the serial port is 7E FF 06 03 00 00 01 EF

7E---Start command;

FF---version information;

06---Data length (excluding checksum);

03---represents the command byte;

00---Whether a response is required [0x01: a response is required, 0x00: no response is required];

00---high byte of the track [DH];

01---The low byte of the track [DL], which represents the first song played;

EF---end command;

(2) For song selection, if you select the 100th song, first convert 100 into hexadecimal, the default is double byte, which is 0x0064. DH=0x00; DL=0x64.

(3) If you choose to play the 1000th song, first convert 1000 into hexadecimal, which is double-byte by default, which is 0x03E8. DH=0x03; DL=0xE8.

(4) Other operations can be deduced in the same way, because hexadecimal is the most convenient operation in the embedded field.

### 5.4.2 Specify the playback volume

(1) The default volume of our system is 30 levels when it is powered on. If you want to set the volume, just send the corresponding command (7E FF 06 06 00 XX XX EF).

(2) For example, if the volume is set to level 15, 15 is converted to hexadecimal as 0x000F, the high 8 bits of DH are 0x00, and the low 8 bits of DL are 0x0F. The actual transmission is: 7E FF 06 06 00 00 0F EF.

### 5.4.3 Specify the playback device

Specify playback device - USB flash drive	7E FF 06 09 00 00 01 EF	
Specify playback device - SD disk	7E FF 06 09 00 00 02 EF	
Specify playback device - AUX	7E FF 06 09 00 00 03 EF	
Specify playback device-FLASH	7E FF 06 09 00 00 04 EF	
Specify playback device - PC	7E FF 06 09 00 00 05 EF	Refers to (card reader, sound card) mode
Specify playback device - SLEEP	7E FF 06 09 00 00 06 EF	

(1) Our module supports 4 types of playback devices by default. Only when the device is online can you specify the device to play. Our software will automatically detect whether the device is online, and the user does not need to worry about it.

(2) See the table below and select the appropriate command to send.

(3) After specifying the device, the module will automatically enter the stop decoding state and wait for the user to specify the track to play. It takes about 200ms from receiving the specified device to completing the initialization of the file system inside the module. Please wait 200ms before sending the command to specify the track.

#### 5.4.4 Play in a specified folder

001xxx.mp3 in the specified folder 01	7E FF 06 0F 00 01 01 EF	
100xxx.mp3 in the specified folder 11 255xxx.mp3 in the specified folder 99	7E FF 06 0F 00 0B 64 EF	
DL Playing in a specified folder 99	7E FF 06 0F 00 63 FF EF	

DL is an extended function we have developed. The default folder naming method is "01" and "11". This method is because our module does not support folder name recognition with Chinese characters. For the stability of the system and the speed of song switching, each folder supports a maximum of 255 songs by default, and supports a maximum of 99 folders. If the customer has special requirements and needs to classify according to English names, we can also achieve this, but the name can only be composed of English names such as "GUSHI" and "ERGE". However, the mp3 file needs to have a prefix, which can be changed from "Have to Love.mp3" to "002 Have to Love.mp3".

(2) For example, to specify the 100xxx.MP3 file in the "01" folder, the command sent by the serial port is: 7E FF 06 0F 00 01 64 xx xx EF

DH: represents the name of the folder. By default, it supports 99 files, i.e., the naming from 01 to 99.

DL: represents the tracks, with a maximum of 255 songs by default, i.e. 0x01-0xFF;

(3) For the sake of module standardization, you must specify both the folder and the file name to lock a file. You can also specify a folder or a file name alone, but this will make file management worse. Specifying folders and specifying tracks supports MP3 and WAV.

(4) The following two screenshots illustrate the designation of folders and file names (divided into two pictures on the left and right):



### 5.4.5 All loop playback instructions

Specify MP3 folder	7E FF 06 12 00 00 01 EF	"MP3" folder, track number is "0001"
	7E FF 06 12 00 00 02 EF	"MP3" folder, track number is "0002"
	7E FF 06 12 00 00 FF EF	"MP3" folder, track number is "0255"
	7E FF 06 12 00 07 CF EF	"MP3" folder, track "1999"
	7E FF 06 12 00 0B B8 EF	"MP3" folder, track number is "3000"

(1) Based on the specified folder and file name, we extend the function of a single folder. The folder name must be is "MP3".

(2) A maximum of 65536 tracks are supported, but due to the operating speed of the file system, as the size of the file increases, The track switching speed will be slowed down accordingly.

(3) The specified file is named as follows:



### 5.4.6 Insert advertisements in the ADVERT folder

Ad breaks	7E FF 06 13 00 00 01 EF	"ADVERT" folder, track number is "0001"
	7E FF 06 13 00 00 02 EF	"ADVERT" folder, track number is "0002"
	7E FF 06 13 00 00 FF EF	"ADVERT" folder, track number is "0255"
	7E FF 06 13 00 07 CF EF	"ADVERT" folder, track number is "1999"
	7E FF 06 13 00 0B B8 EF	"ADVERT" folder, track number is "3000"

(1) We support inserting other tracks during the song selection process, so as to meet the need for background music. The need to insert advertisements during the playing of music.

(2) After sending the 0x13 command, the system will store the IDV3 information of the currently playing track, and then play the command again. After the inserted track is finished playing, the system will return to the saved breakpoint to continue Play until it is finished.

(3) The format is to create a folder named "ADVERT" in the device and store the files to be inserted. The track setting is "0xxx + track name.MP3/WAV".

(4) If the system is currently in a paused or stopped state, sending an interrupt command will not receive a response and will There is an error message returned. If you are in the process of inserting other tracks, you can continue to insert other tracks, but after the playback is completed, Or return to the IDV3 information stored for the first time.

### 5.4.7 A single folder supports 1,000 tracks

Support 1000 songs	7E FF 06 14 00 10 FF EF	Specify folder "01" and track "0255"
	7E FF 06 14 00 17 CF EF	Specify the folder "01" and the track "1999"
	7E FF 06 14 00 C0 01 EF	Specify folder "12" and track "0001"
	7E FF 06 14 00 C0 FF EF	Specify the folder "12" and the track "0255"
	7E FF 06 14 00 C7 CF EF	Specify the folder "12" and the track "1999"

(1) In response to many customers' requirements for 10 folders, each of which can manage 1,000 tracks, we specially

This instruction is added for users to call, and the detailed description is as follows:

The command byte of the serial port is 0x14;

The parameter is two bytes. If the folder is specified as "12", the track is "1999";

Serial port data: 7E FF 06 14 00 C7 CF EF;

0xC7 and 0xCF are parameters, which together are 0xC7CF, a total of 16 bits, of which the high 4 bits represent the text.

The lower 12 bits represent the file name, and 7CF represents

1999, that is, tracks with the file prefix "1999".

### 5.4.8 All loop playback instructions

Loop playback starts	7E FF 06 11 00 00 01 EF	Loop all tracks
Loop playback stop	7E FF 06 11 00 00 00 EF	Stop looping a track

(1) To meet the requirements of looping the tracks in the root directory, we add this control instruction 0x11.

(2) During the loop playback, you can operate the normal operations of play/pause, previous song, next song, and volume adjustment.

Including EQ and so on.

(3) After the loop playback starts, the module will continuously play the tracks in the device in the physical order in which they are stored.

After finishing one time, it will continue to play the other side until receiving the playback completion, pause and other instructions.

### 5.4.9 Single song loop play command

Loop playback starts	7E FF 06 08 00 00 01 EF	Play the first song on loop
Loop playback stop	7E FF 06 08 00 00 02 EF	Play the second song on loop

(1) To meet the requirements of looping a single song, we improved the control instruction 0x08.

(2) During the loop playback, you can operate the normal operations of play/pause, previous song, next song, and volume adjustment.

Including EQ, etc., and the state is still looping, you can trigger playback or enter sleep by specifying a single song

Turn off loop playback.



#### 5.4.10 Single song loop play command

Now Playing	7E FF 06 42 00 00 01 EF	Now Playing
Pause playback	7E FF 06 42 00 00 02 EF	Paused during playback
Stop Playing	7E FF 06 42 00 00 00 EF	Playback completed
Currently in sleep state	7E FF 06 42 00 00 08 EF	No device is online or is set to sleep.

(1) The module will have four states open to the user during the decoding process. The user can query the current state of the module through the command Previous state.

(2) Playback pause means that a song is being played and a command is sent manually to pause the playback.

(3) Playback stop means that when a song is finished playing, the module stops playing.

#### 5.4.11 Playback stop command

Stop playing ads	7E FF 06 15 00 00 00 EF	Stop being an ad and return to background music
Stop Playing	7E FF 06 16 00 00 00 EF	Stop software decoding

(1) During the playback of the module, we have two ways to stop. One is to stop the current interstitial advertisement and return to the current One is to continue playing the background music at the previous breakpoint, and the other is to stop all playback, including the background music.

(2) If an interstitial advertisement is currently playing, the chip will stop all playback tasks if the stop command 0x16 is sent.

#### 5.4.12 Loop playback in a specified folder

Loop playback of specified folder	7E FF 06 17 00 00 02 EF	Specify 02 folder to play in loop
	7E FF 06 17 00 00 01 EF	Specify 01 folder to play in loop

(1) The folder naming format must be "01" to "99" and cannot exceed 99.

(2) After specifying a folder, the video will play in a loop within the specified folder and will not stop unless a stop command is sent.

Order and so on.

#### 5.4.13 Randomly play device files

Shuffle	7E FF 06 17 00 00 02 EF	All files are played randomly and in a loop
---------	-------------------------	---

(1) This command randomly plays the voice files stored in the device in physical order, regardless of the device.

Check whether there is a folder in it, and the first voice file played must be the first voice file in the device.

#### 5.4.14 Set the current track to loop

Loop playback of specified files	7E FF 06 19 00 00 00 EF	Single song loop playback on
	7E FF 06 19 00 00 01 EF	Single song loop play off

(1) Sending this command during playback will loop the current track.

If the chip is in stop state, it will not respond to this command.

(2) If you want to turn off the single song loop playback, send a close command, which will stop the current song after it is played.

Then stop.

#### 5.4.15 Combined broadcast

Combined broadcast	7E FF 08 20 00 01 01 01 02 EF	7E FF XX 20 00 XX XX EF
--------------------	-------------------------------	-------------------------

(1) Combined broadcast is mainly used in shopping malls, shopping centers and public transportation stations. TD5580A supports up to 30  
If the user has higher requirements, please contact our online customer service.

(2) For example, in the instruction 7E FF 08 20 00 01 01 01 02 EF, 08 is the LEN of the instruction, which means the LEN bits of the instruction are removed from the start and end of the frame.

The length after the bundle and checksum, for example, 001, 003, 005 in multicast folder 01 and 005, 003, 001 in folder 02,

You can send 7E FF 10 20 00 01 01 01 03 01 05 02 05 02 03 02 01 EF.

(3) If the customer needs a more perfect combination of broadcasts, the customer needs to trim the audio file, such as

The blank content at the beginning and end is trimmed to achieve a more perfect connection effect.

#### 5.4.16 Turning the DAC on and off

Setting up the DAC	7E FF 06 1A 00 00 00 EF	Turn on DAC
	7E FF 06 1A 00 00 01 EF	DAC off (high impedance)

(1) The chip can turn off the DAC at any time. If the voice is currently playing and the DAC is turned off, the chip will

The chip is powered on by default, and the DAC is turned on only when it is set to off.

If you need to open it again, you need to open the DAC through instructions.

## 6. ADKEY button description

Button resistance		operate	Functional Description
K1	0R	Short press	Short press to play/pause, long press to switch music devices
K2	6.8K	Short press/Long press	Short press to play the previous file, long press to reduce the volume
K3	10K	Short press/long	Short press to play the next file, long press to increase the volume

press Note: Please refer to the circuit for key operation. If the key function needs to be changed, please contact online customer service for customization.

## 7. Customized functions can be developed

If the above standard functions cannot meet customer needs, the chip can be developed and customized with some additional functions.

Yes. If you need to develop customized functions, you need to list the required functions first and then send them to our sales staff.

The project will be evaluated, and once the evaluation is passed, engineers will contact the customer to develop the corresponding functions.

Some common features:

(1) Increase the number of buttons and their corresponding functions, such as adding one-to-one functions to three buttons, adding a loop control function, etc., and changing the default control mode when powered on, for example, the chip starts playing songs in the memory in a loop when powered on.

(2) Add LED lights to indicate the status, for example: add 2 LED lights, play the first voice, the first light is on, play the second voice, the second light is on.

(3) Add certain instructions, such as deletion instructions or formatting instructions.

## 8. FAQ:

Q: Is it necessary to add an amplifier to the chip-driven speaker?

A: Yes, this chip can only drive headphones directly. If you need to connect an external speaker, even if the power is very low, you will need an external amplifier.

Q: How to quickly update the audio content on SPI-Flash?

A: The chip supports updating the SPI-Flash content through the USB interface. Connect one end of the USB male-to-male connector to the computer and the other end to the USB port on the board. After the connection, a drive letter will pop up on the computer, and the content can be updated.

Q: Serial port commands don't work?

A: The DEMO board can work after it is powered on. You need to connect the TX and RX of the serial port tool to the RX and TX of the DEMO board respectively, check whether the serial port working voltage is 3.3V, and check whether the baud rate is 9600.