目 录

1. 概述 3

1.1 简介 3
1.2 功能 3
1.3 应用 3

2. 芯片功能框架说明 4

2.2 管脚说明 5

3. 串口通讯协议 6

3.1 通讯格式 6
3.2 通讯指令 7
3.3 模块返回的数据 9
3.3.1 模块上电返回的数据 9
3.3.2 曲目播放完毕返回的数据 9
3.3.3 模块应答返回的数据 10
3.3.4 模块错误返回的数据 10
3.4 串口指令详解 11
3.4.1 指定歌曲播放指令[可以直接参考 3.4.7] 11
3.4.2 指定音量播放指令 11
3.4.3 全部循环播放指令 12
3.4.4 单曲循环播放指令 12
3.4.5 组合播放功能指令 12
3.4.6 带音量参数的指令播放 13
3.4.7 带指定文件夹的曲目播放 13
3.4.8 指定文件夹的曲目单曲循环播放 13
3.4.9 指定文件夹的曲目开始循环播放 13

4. 参考电路 14

4.1 串行接口 14
4.2 按键接口 15
4.3 外接单声道功放 18
4.4 USB 更新语音说明 19
4.5 用户使用空白的 FLASH 说明 21
4.6 组合播放的参考例程 22
4.7 模块的硬件说明 23

4.8 模块尺寸 23

5. 注意事项.....24

5.1 GPIO 的特性.....24
5.2 应用中的注意点.....25
5.3 串口操作.....26
5.3.1 串口操作流程.....26
5.3.2 串口编程参考的说明.....27
5.3.3 串口编程需要适当延时的注意点.....27
5.3.4 校验的重要说明.....27

6. 免责声明.....28

1. 概述

1.1 简介

Msk330-16P 是一个提供串口的语音模块，完美的集成了 MP3、WAV 的硬解码。同时软件支持工业级别的串口通信协议，以 SPIFLASH 作为存储介质，降低了成本同时也增加了产品的稳定性。通过简单的串口指令即可完成播放指定的语音，以及如何播放语音等功能，无需繁琐的底层操作，使用方便，稳定可靠是此款产品的最大特点。另外该模块也是深度定制的产品，集成了 3W 的功放，直接外接喇叭即可放音。同时模块支持 10 段语音的指定触发播放。

1.2 功能

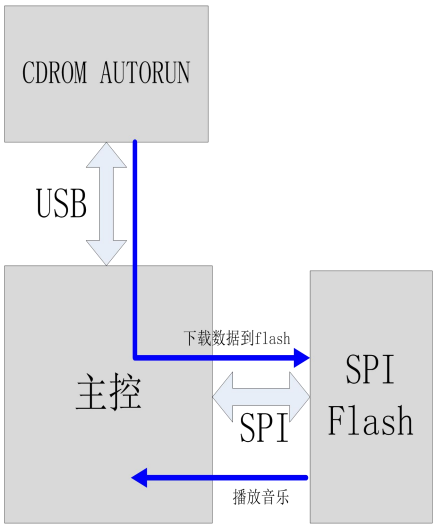
1、支持采样率(KHz):8/11.025/12/16/22.05/24/32/44.1/48
2、24 位 DAC 输出，动态范围支持 90dB，信噪比支持 85dB
3、最大支持 16M 字节的 SPIFLASH。例如 W25Q16[2M 字节]、W25Q128[16M 字节]
4、多种控制模式，并口控制模式、串口模式、AD 按键控制模式
5、Microusb 接口更新语音文件，无需安装任何软件。支持 XP 和 WIN7 系统。
6、支持组合播放功能，可以实现报时、报温度，在一定程度上可以替代一些昂贵的 TTS 方案
7、30 级音量可调，5 级 EQ 可调
8、自带 3W 的功放，直接外接喇叭即可完成放音
9、支持 10 段语音的触发播放，IO 检测的方式，所以适合碳膜按键等等场合

1.3 应用

1、 车载导航语音播报
2、 公路运输稽查、收费站语音提示；
3、 火车站、汽车站安全检查语音提示；
4、 电力、通信、金融营业厅语音提示；
5、 车辆进、出通道验证语音提示；
6、 公安边防检查通道语音提示；
7、 多路语音告警或设备操作引导语音；
8、 电动观光车安全行驶语音告示；
9、 机电设备故障自动报警；
10、消防语音报警提示；

11、自动广播设备，定时播报

2. 芯片功能框架说明



模块选用的是 SOC 方案，集成了一个 16 位的 MCU，以及一个专门针对音频解码的 aDSP，采用硬解码的方式，更加保证了系统的稳定性和音质。小巧的封装尺寸更加满足嵌入其它产品的需求

2.1 硬件参数

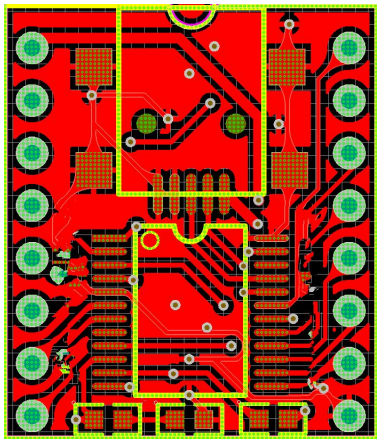
名称	参数
MP3文件格式	1、支持所有比特率11172-3和 ISO13813-3 layer3音频解码
	2、采样率支持(KHZ):8/11.025/12/16/22.05/24/32/44.1/48
	3、支持 Normal、Jazz、Classic、Pop、Rock 等音效
USB 接口	2.0标准
UART 接口	标准串口，TTL 电平,波特率可设[用户不可设]
输入电压	3.3V-5V[7805后级串一个二极管为最佳]
额定电流	10MA[静态]
低功耗电流	<200uA
功放功率	负载4欧姆，支持3W
尺寸	25(长)*20(宽)[单位:mm]
工作温度	-40度 -- 80度

湿度	5% ~ 95%

2.2 管脚说明

引脚序号	引脚名称	功能描述	备注
1	VDD	电源输入[正极]	最大5.2V
2	SPK1	外接单声道喇叭	不可超过4欧姆3W
3	SPK2	外接单声道喇叭	不可超过4欧姆3W
4	DAC-R	外接功放、耳机	
5	DAC-L	外接功放、耳机	
6	BUSY	播放状态脚	播放时输出低
7	TX	UART 串行数据输出	3.3V 的 TTL 电平
8	RX	UART 串行数据输入	3.3V 的 TTL 电平
9	GND	地[负极]	系统地
10	IO1	通用输入输出口1	
11	IO2	通用输入输出口2	
12	IO3	通用输入输出口3	
13	IO4	通用输入输出口4	
14	IO5	通用输入输出口5	
15	IO6	通用输入输出口6	
16	IO7	通用输入输出口7	

备注：模块内置单声道功放。另外模块有白点的为第 2 脚和第 3 脚



模块分布说明:

(1)、管脚 1--8，从右边上往下数

(2)、管脚 9--16，从左边依次下往上数

(3)、功放芯片[4 欧姆、3W]

(4)、主控芯片[SSOP24 封装]

(5)、USB 接口座子[兼容 Miniusb 和 Microusb]

(6)、自带了一个 SPI flash

用户可以用镊子直接短路第 9 脚和第 10 脚，来触发模块播放。作为测试模块是否正常的手段。

3. 串口通讯协议

串口作为一种在控制领域常用的通信，我们进行了工业级别的优化，加入的帧的校验、重发、错误处理等措施，大大加强通信的稳定性和可靠性，同时可以在此基础上扩展更加强大的 RS485 进行组网功能，串口的通信波特率可自行设置，默认为 9600

3.1 通讯格式

支持异步串口通讯模式,通过串口接受上位机发送的命令

通讯标准:9600 bps
数据位 :1
校验位 :none
流控制 :none

格式：$S VER Len CMD Feedback para1 para2 checksum $0		
$S	起始位0x7E	每条命令反馈均以$开头,即0x7E
VER	版本	版本信息
Len	len 后字节个数	校验和不计算在内
CMD	命令字	表示具体的操作,比如播放/暂停等等
Feedback	命令反馈	是否需要反馈信息,1反馈,0不反馈
dat	参数	和前面的 len 相关联，不限制长度
checksum	校验和[占两个字节]	累加和校验[不计起始位$]
$0	结束位	结束位0xEF

举个例子，如果我们指定播放 SPIFLASH，就需要发送:7E FF 06 09 00 00 04 FF dd EF
数据长度为 6 ,这 6 个字节分别是[FF 06 09 00 00 04] 。不计算起始、结束、和校验。然后对得到的结果进行累加，再用 0 减，即“0-checksum=校验数据”。如果这里不明白，可以参考我们的“MSK330 辅助说明文档”。另外用户也可以直接忽视校验，参考我们的 5.3.4 章节说明。

3.2 通讯指令

1、直接发送的指令，不需要返回参数

CMD 命令(指令)	对应的功能	参数(16位)
0x01	下一曲	
0x02	上一曲	
0x03	指定曲目(NUM)	1-255
0x04	音量+	
0x05	音量-	
0x06	指定音量	0-30
0x07	指定 EQ0/1/2/3/4/5	保留
0x08	单曲循环指定曲目播放	1-255
0x09	指定播放 SPIFLASH	FLASH
0x0A	进入睡眠 -- 低功耗	功耗维持在10MA
0x0B	保留	
0x0C	模块复位	
0x0D	播放	
0x0E	暂停	
0x16	停止	
0x21	组合播放	见下面的详细说明
0x22	带音量的指定曲目播放	见下面的详细说明
0x30	指定文件夹的文件播放	见一下详细说明
0x31		
0x32		
0x33		

2、查询系统的参数

CMD 命令详解(查询)	对应的功能	参数(16位)
0x3C	STAY	
0x3D	STAY	
0x3E	STAY	
0x3F	发送初始化参数	0x1F(低5位每位代表一个文件夹)
0x40	返回错误，请求重发	
0x41	应答	
0x42	查询当前状态	
0x43	查询当前音量	
0x44	查询当前 EQ	保留
0x45	查询当前播放模式	该版本保留此功能
0x46	查询当前软件版本	该版本保留此功能
0x49	查询 FLASH 的总文件数	5个文件夹的总文件数
0x4D	查询 FLASH 的当前曲目	返回文件夹和曲目指针
0x60	查询当前在线的文件夹	
0x61	查询当前文件夹指针	
0x62	查询指定文件夹的曲目总数	

3.3 模块返回的数据

模块在关键地方均会有数据返回。供用户掌控模块的工作状态

- 模块上电初始化成功的数据
- 模块播放完当前曲目的数据
- 模块成功接收到指令返回的 ACK(应答)
- 模块接收一帧数据出错[包括数据没收完整、校验出错两种情况]
- 模块在繁忙时，有数据过来，模块会返回忙的指令

3.3.1 模块上电返回的数据

(1)、模块上电，需要一定的时间初始化，这个时间是需要根据 SPIFLASH 设备的文件多少决定的，一般情况在小于 500ms 这个时间。如果超过这个时间模块的初始化数据还没有发送出来，说明模块初始化出错，请检查硬件的连接

(2)、模块初始化返回的数据为当前的有效文件夹,譬如返回 7E FF 06 3F 00 00 03 xx xx EF

==>其中 0x03 代表的是 FOLDER1 和 FOLDER2 这两个区域有有效语音文件

如果是 0x07 代表的是 FOLDER1 和 FOLDER2、FOLDER3 都有语音文件

实际上，5 个文件夹均占据一个字节的一位。FOLDER 占据最低位

(3)、MCU 必须等待模块初始化指令发出之后才能发送相应的控制指令，否则发送的指令模块将不予处理。同时还会影响模块的正常初始化。

3.3.2 曲目播放完毕返回的数据

FLASH 播放完第1曲	7E FF 06 3E 00 01 01 xx xx EF	FOLDER1的第1曲播放完
FLASH 播放完第2曲	7E FF 06 3E 00 02 02 xx xx EF	FOLDER2的第2曲播放完

1、争很多的触发型的播放需求，我们模块更正为播放一曲之后自动进入停止状态。如果用户需要此类应用。只需要指定曲目播放即可。这样，曲目播放完毕会自动停止，等待指令

2、另外我们专门开辟一个 IO 作为解码和暂停的状态指示。请参见第 5 脚

- 、播放状态输出低电平[很多功放有静音脚，可以通过此 IO 直接进行控制]

■ 、播放暂停状态，输出高电平。模块睡眠状态。也是低电平

3、模块通电之后，初始化正常 ，模块会自动进入设备播放状态。并且停止解码，等待用户发送播放的相关指令

4、另外用户在指定设备之后，需要等待 200ms 的时间，再发送指定的曲目，因为一旦指定曲目之后，系统会对指定的设备进行文件系统的初始化，如果立刻发送指定的曲目命令，会导致模块接收不到。

3.3.3 模块应答返回的数据

(1)、为了加强数据通信之间的稳定性，我们增加了应答处理，ACKB 字节就是设置是否需要回复应答。这样做的好处是保证每次通信都有握手信号，收到应答就表示 MCU 发送的数据，模块已经成功收到，马上处理。

(2)、对于一般的应用，客户可以自由选择，不加这个应答处理也是可以的。

3.3.4 模块错误返回的数据

返回忙	7E FF 06 40 00 00 01 xx xx EF	模块在文件系统初始化时
当前处于睡眠模式	7E FF 06 40 00 00 02 xx xx EF	睡眠模式只支持指定设备
串口接收错误	7E FF 06 40 00 00 03 xx xx EF	串口一帧数据没接收完毕
校验出错	7E FF 06 40 00 00 04 xx xx EF	和校验出错
指定文件超范围	7E FF 06 40 00 00 05 xx xx EF	文件的指定超过设定的范围
未找到指定的文件	7E FF 06 40 00 00 06 xx xx EF	指定为文件没有被找到
数据不符合规则	7E FF 06 40 00 00 08 xx xx EF	如最小为1的地方，发送为0

(1)、为了加强数据通信之间的稳定性，我们增加了数据错误处理机制。模块收到不符合格式的数据，均会有信息反馈出来

(2)、在环境比较恶劣的情况下，强烈建议客户处理此命令。如果应用环境一般，可以不用处理。

(3)、模块返回忙，基本上是模块上电初始化的时候才会返回，因为模块需要初始化文件系统

(4)、模块上电之后，进入的是设备状态，设备是 SPIFLASH。如果 SPIFLASH 不在线的话，会自动进入睡眠状态。

(5)、只要参考我们给出的测试 SDK 程序，移植里面的串口操作部分，就不会出现校验出错，在这里强烈建议用户使用我们给出的校验方式。因为谁都不能保证数据的传输不会出错。

(6)、文件指定部分出错，请参考下面的详解

3.4 串口指令详解

以下我们对关键的地方进行详细的说明:

- 指定曲目播放
- 指定播放的音量
- 指定播放的设备
- 全部循环播放指令
- 组合播放功能[亮点]
- 带音量参数的指定曲目播放

3.4.1 指定歌曲播放指令[可以直接参考 3.4.7]

我们给出的指令是支持指定曲目播放的，歌曲的选择范围为 0~255.其实是可以支持更多的，因为涉及到文件管理的原因，支持过多的歌曲，会导致系统操作缓慢，一般的应用也不需要支持这么多的文件。如果客户有非常规的应用，请事前和我们沟通。

(1)、例如选择第一首歌播放，串口的发送部分 7E FF 06 03 00 00 01 FF E6 EF

7E --- 起始命令

FF --- 版本信息

06 --- 数据长度(不包含校验)

03 --- 代表产品编号

00 --- 是否需要应答[0x01:需要应答，0x00:不需要返回应答]

00 --- 曲目的高字节[DH]

01 --- 曲目的低字节[DL],这里代表的是第一首歌播放

FF --- 校验的高字节

E6 --- 校验的低字节

EF --- 结束命令

(2)、对于选曲，如果选择第 100 首，首先将 100 转化为 16 进制,默认为双字节,就为 0x0064。

DH = 0x00 ; DL = 0x64

(3)、其它的操作依次类推即可，因为在嵌入式领域采用 16 进制是最为方便的一种操作。

3.4.2 指定音量播放指令

(1)、我们系统上电默认的音量为 30 级，如果要设置音量的话,直接发送相应的指令即可
(2)、例如指定音量为 15 级,串口发送的指令:7E FF 06 06 00 00 0F FF D5 EF
(3)、DH = 0x00 ; DL = 0x0F ，15 转化为 16 进制为 0x000F。可以参照播放曲目部分的说明

3.4.3 全部循环播放指令

0x11	循环播放	[1:循环播放][0:停止循环播放]

1、争对一些需要循环播放根目录下面曲目的要求，我们加多这一条控制指令 0x11。

循环播放开始	7E FF 06 11 00 00 01 xx xx EF	循环播放所有曲目
循环播放停止	7E FF 06 11 00 00 00 xx xx EF	停止循环播放曲目

2、在循环播放的过程中，可以正常的操作播放/暂停，上一曲、下一曲、音量调节，包括 EQ 等等
3、循环播放开始之后，模块会不停的播放设备里面的曲目，按照存储的物理顺序。播完一遍之后会继续再播放一边，直到接收到播放完成，或者暂停等等指令

3.4.4 单曲循环播放指令

循环播放开始	7E FF 06 08 00 00 01 xx xx EF	循环播放第一曲
循环播放停止	7E FF 06 08 00 00 02 xx xx EF	循环播放第二曲

(1)、争对一些需要单曲循环播放的要求，我们改进这一条控制指令 0x08。按照的是存储的物理顺序指定，这点请用户注意。
(2)、在循环播放的过程中，可以正常的操作播放/暂停，上一曲、下一曲、音量调节，包括 EQ 等等并且状态仍然是循环播放.可以通过指定单曲触发播放或者进入睡眠来关闭循环播放状.

3.4.5 组合播放功能指令

组合播放	7E FF 09 21 00 05 01 02 03 04 FE C8 EF	播放5、1、2、3、4
组合播放	7E FF 0C 21 00 05 01 02 03 04 06 07 08 FE B0 EF	播放5、1、2、3、4、6、7、8

(1)、很多应用场合需要用到 TTS 的功能，譬如报时，报温度，报金额等等应用，如果用户拿我们的模块模仿简单的 TTS 功能的话，可能会在效果上面大打折扣，矛盾点就是在语音和语音之间的延时。直接按照一个一个文件的指定播放的话，会存在语音和语音之间的延时，这样是不能接受的。因此我们增加了组合播放的功能，同时支持指定播放 12 个语音，按照串口发送的顺序依次播放。
(2)、如果发送 7E FF 0C 21 00 05 01 02 03 04 06 07 08 FE B0 EF 这一帧数据，我们分析一下

CMD= 0x21 --- 查阅指令表

LEN = 0x0c = 12 个字节 --- FF 0C 21 00 05 01 02 03 04 06 07 08[去掉红色标记的字节，就是曲目的顺序]

模块会一次播放第 5 曲、第 1 曲、2、3、4、6、7 这 7 首曲目。播放完毕就停止

(3)、在组合播放的过程中，是允许用户进入播放暂停和设置音量，但是不允许指定上下曲。如果用户对组合播放的要求比较高的话，请用户对音源自行编辑一下，去掉音源头和尾的一些静音。这样就可以减少语音和语音之间的延时，可以采用“Adobe Audition CS5.5”或者“GoldWave.exe”等等专业音频软件制作。

(4)、有了这个功能，就可以很方便的实现“欢迎光临，现在时间是 13 年 12 月 24 日”这样的灵活播报方式，大大提高了产品的竞争力。

3.4.6 带音量参数的指令播放

带音量播放	7E FF 06 22 00 1E 01 xx xx EF	30级音量播放第1曲
	7E FF 06 22 00 0F 02 xx xx EF	15级音量播放第2曲

(1)、争对一些用户希望，对不同的语音设置不同的音量进行播放，如果按照以前的老方法，就是先设置完音量，再指定曲目播放，这样操作繁琐，不方便。特此我们增加此条指令 0x22

(2)、具体的操作可以参考上面给出的两条测试指令。

3.4.7 带指定文件夹的曲目播放

指定区域和文件播放	7E FF 06 30 00 01 01 FE C9 EF	FOLDER1的第1曲
	7E FF 06 30 00 02 01 FE C8 EF	FOLDER2的第1曲

(1)、我们的模块可以将 FLASH 的区域分为 5 块，可以针对里面的任何一块进行操作

(2)、具体的操作可以参考上面给出的两条测试指令。

3.4.8 指定文件夹的曲目单曲循环播放

指定文件夹曲目循环播放	7E FF 06 31 00 03 03 FE C4 EF	FOLDER3的第3曲循环播放
	7E FF 06 31 00 01 03 FE C6 EF	FOLDER1的第3曲循环播放

(1)、用户可以对 5 个文件夹里面的任意一个文件进行指定单曲循环播放，请参考上面的参考指令

(2)、用户可以发送停止指令来停止单曲循环播放。

3.4.9 指定文件夹的曲目开始循环播放

指定文件夹循环播放	7E FF 06 32 00 03 01 FE C5 EF	FOLDER3的第1曲循环播放

(1)、用户可以对 5 个文件夹里面的任意一个文件夹循环播放，请参考上面的参考指令。

03 指定的文件夹为 FOLDER3

01 指定文件夹的第一曲开始，如果这里为 02。那么从第 2 曲开始循环播放此文件夹

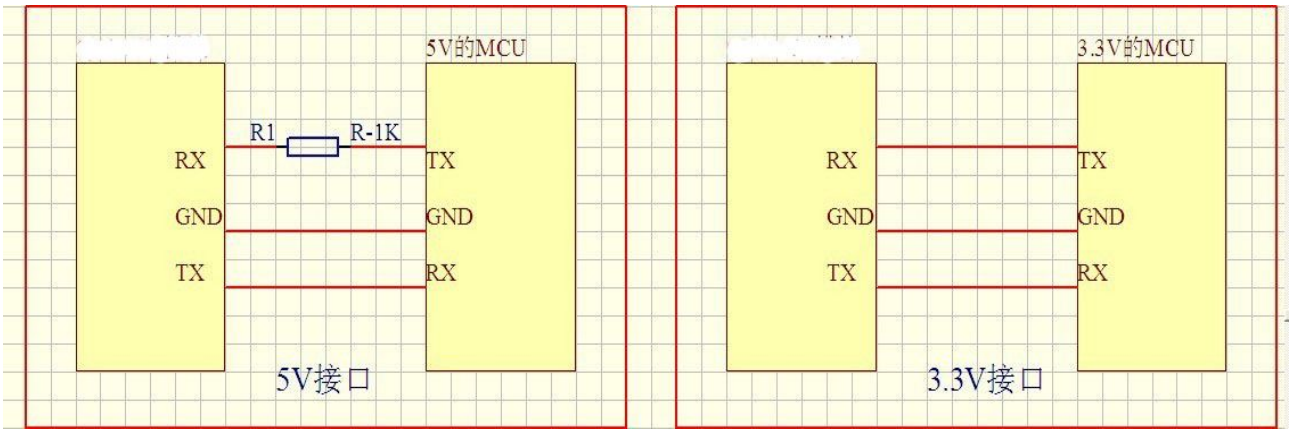
(2)、用户可以发送停止指令来结束循环播放。

4. 参考电路

争对模块的应用，我们提供了详细的设计参考，让您可以更快的上手体验到该模块的强大功能

- 串行通信接口，波特率默认 9600，可以根据客户的要求修改
- 外部的 IO 按键的功能可以按照客户需求订制
- 外部单声道功放参考电路

4.1 串行接口



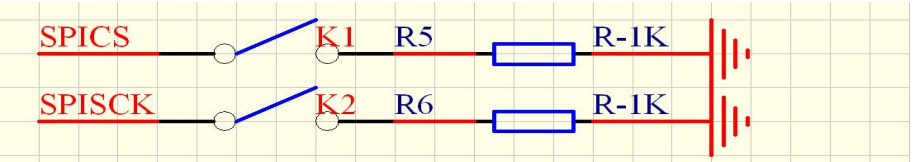
模块的串口为 3.3V 的 TTL 电平，所以默认的接口的电平为 3.3V。如果系统是 5V。那么建议在串口的对接接口串联一个 1K 的电阻。这样足以满足一般的要求，如果应用于强电磁干扰的场合，请参考“注意事项”的说明。模块在 5V 和 3.3V 的系统中均正常的测试过，一切正常。均在采用的是直连的方式，并没有串 1K 的电阻。

4.2 按键接口

模块我们采用的是 IO 按键的方式，取代了 AD 键盘的接法，这样做的好处是充分利用了 MCU 越来越来越多的 GPIO。设计繁琐但不简单，我们模块默认配置 10 个按键的功能分配，可以在任何恶劣的场合随意的控制，甚至也可以作为与 MCU 的通信接口。我们的按键分配 4 种不同类型的功能，根据两个电阻的对地选择，请联系技术支持。

- 一对一触发播放，可打断
- 一对一触发播放，电平保持可循环
- 一对一触发播放，不可打断
- 标准的播放功能，如上下曲、播放暂停等等

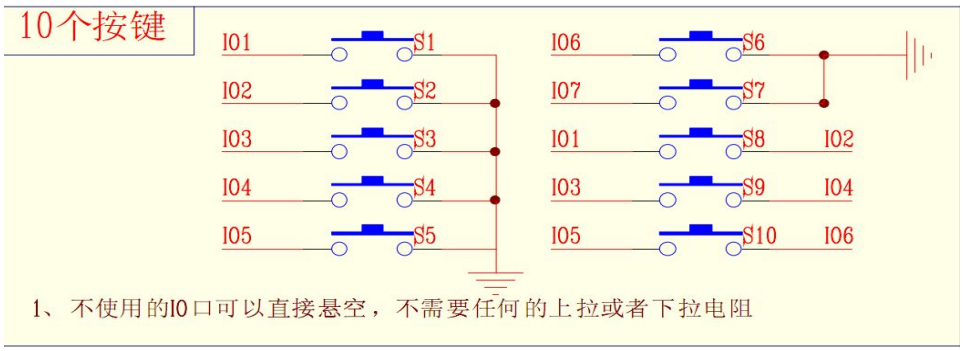
丝印 1 对应 K1　丝印 2 对应 K2[详见 PCB]



K1	K2	对地短路为0,悬空为1
1	1	一对一触发播放[可打断]
0	1	按键抬起停止播放
1	0	一对一触发播放[不可打断]
0	0	标准MP3功能

备注:标准MP3功能包含上下曲和音量加减以及暂停和停止

上表为选择 10 个按键的功能逻辑表。模块出厂默认为[一对一触发播放可打断]



模块留出来的按键为 S1--S7.其它的 S8--S10 需要用户自行按照上图的原理进行接线

(1)、一对一触发功能[可打断]

一对一触发	短按	长按	按着不松	按键抬起
S1	第1段[FOLDER1]			
S2	第2段[FOLDER1]			
S3	第3段[FOLDER1]			
S4	第4段[FOLDER1]			
S5	第5段[FOLDER1]			
S6	第6段[FOLDER1]			
S7	第7段[FOLDER1]			
S8	第8段[FOLDER1]			
S9	第9段[FOLDER1]			
S10	第10[FOLDER1]			
备注:此为一对一触发播放，可打断[对地短路触发80ms 有效]				

(2)、按键抬起停止播放功能

按键抬起停止	短按	长按	按着不松	按键抬起

S1		第1段循环[FOLDER1]		停止
S2		第2段循环[FOLDER1]		停止
S3		第3段循环[FOLDER1]		停止
S4		第4段循环[FOLDER1]		停止
S5		第5段循环[FOLDER1]		停止
S6		第6段循环[FOLDER1]		停止
S7		第7段循环[FOLDER1]		停止
S8		第8段循环[FOLDER1]		停止
S9		第9段循环[FOLDER1]		停止
S10		第10段循环[FOLDER1]		停止
备注:此为抬起停止，长按循环播放[对地短路触发800ms 有效]				

(3)、一对一触发不可打断

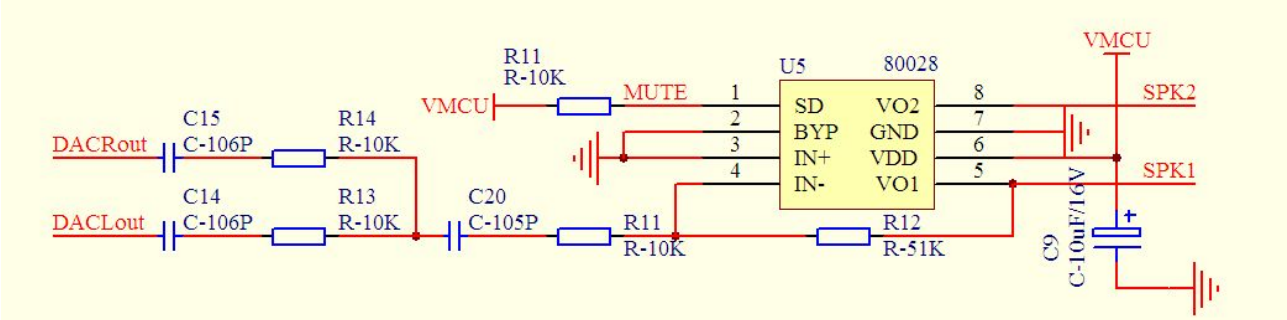
触发不打断	短按	长按	按着不松	按键抬起
S1	第1段不可打断[FOLDER1]			
S2	第2段不可打断[FOLDER1]			
S3	第3段不可打断[FOLDER1]			
S4	第4段不可打断[FOLDER1]			
S5	第5段不可打断[FOLDER1]			
S6	第6段不可打断[FOLDER1]			
S7	第7段不可打断[FOLDER1]			
S8	第8段不可打断[FOLDER1]			
S9	第9段不可打断[FOLDER1]			
S10	第10段不可打断[FOLDER1]			
备注:此为一对一触发播放，不可打断[对地短路触发80ms 有效]				

(4)、标准播放功能

播放功能按键	短按	长按	按着不松	按键抬起
S1	下一曲		音量+	

S2	上一曲		音量-	
S3	播放暂停			
S4	低功耗			
S5	停止			
S6	音量+		音量+	
S7	音量-		音量-	
S8	循环播放			
S9	随机播放			
S10	按下播 FOLDER1第一曲		抬起播放 FOLDER1第二曲	
备注:此为标准的 MP3功能				

4.3 外接单声道功放



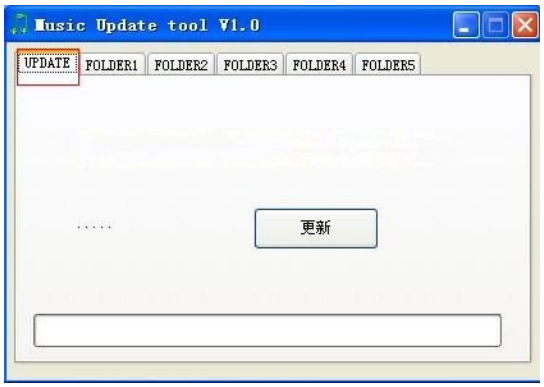
这里功放我们采用的是 8002，具体参数请参考 IC 的 datasheet。应用于一般场合足以，如果追求更高的音质，请客户自行寻找合适的功放。

4.4 USB 更新语音说明

我们的模块可以使用手机充电线直接更新语音，方便、灵活。我们的优势如下

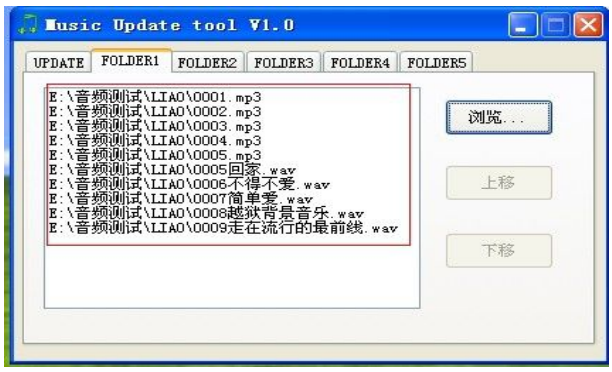
- 可以按照客户的要求，更正下载语音的窗口信息
- 无需安装任何软件，直接更新，也不需要专用下载器
- 对音质无任何压缩和损坏，保证更高的音质体验

1、给模块供电，插上我们的手机充电线，称之为 MicroUSB 线。电脑会弹出如下界面



2、选中“FOLDER1”文件夹，并且点击浏览，就会弹出一个加载语音的窗口，如下图

3、此时选中需要加载的语音，点击“打开”就添加在软件中了。



4、最后，回到“UPDATE”界面，点击“更新”按钮，会出现如下界面

从左至右依次 3 个窗口，最后一个窗口就代表更新完毕，直接关闭此窗口，拔掉 USB 线就可以了

5、目前我们测试过的 SPIFLASH 包括华邦、GD、旺宏等市场上量最大的 FLASH。所以 FLASH 的兼容性不存在问题。我们测试过的 FLASH。最大为 16M 字节[W25Q128],均是没问题。

6、由于我们更新的窗口的更新进度条最大只支持 8M 字节，所以使用 16M 字节的 FLASH 时，会出现如下界面，请用户朋友不用担心，这是正常的

虽然进度条没变，但是还是在更新

7、更新的说明：

(1)、我们的更新窗口预留了 5 个区域进行更新，分别命名为 FOLDER1--FOLDER5.此版本支持 5 个区域的指定播放操作

(2)、FOLDER1--FOLDER5 这 5 个区域并没有分配空间，是可伸缩的。譬如，我只是用 FOLDER1，不使用其它的 4 个文件区域，那这 4 个文件区域会占空间吗？会的，占用不到 10 个字节,可忽略。

(3)、每一个 FOLDER 区域最多支持 255 段语音[在空间允许的情况下]。单个语音的大小是没有要求的。目前 FLASH 最大支持 128Mbit。也就是 16M 字节的语音存储，足以满足大部分的应用

(4)、如果用户对音质要求不高，可以自行选用语音压缩软件，对 MP3 或者 WAV 文件进行压缩，以减少文件所占的存储空间。为了体现我们产品的高音质，不建议用户这么使用

(5)、USB 更新语音的时间，是根据语音文件大小决定的，语音越大，更新的时间越慢。

(6)、一旦更新过语音之后，之前保存的语音将会全部删除。

7、备注：

(1)、我们的模块无需安装任何软件，插入电脑弹出的软件实际是烧录至 SPIFLASH。占 200kb 的空间。很小，所以用户可以直接忽略。

(2)、用户如果需要自行换上自己的 SPIFLASH。请向我司技术支持索取更新的固件。

(3)、如果电脑第一次使用我们的模块，刚插入电脑时，会需要一定的时间，因为此时电脑会自动的对我的模块进行枚举，确定身份等等操作。可能需要 1 分钟左右，直至弹出更新窗口。

4.5 用户使用空白的 FLASH 说明

用户在调试的过程中，会按照自己的需求更换 flash 的大小来满足自己的需求，这样就需要以下三个步骤来完成 FLASH 的替换。

- 问我们获取 FLASH 的更新固件
- 通过 USB 接口对空白的 FLASH 进行固件的烧录
- 拔掉 USB 线缆，再通电就可正常使用

1、我们的固件分为 4 个部分，下面详细解释一下

用户根据自己的 FLASH 大小，来选择更新的固件。固件的大小都是一样的。

实际上是通过"mydown.bat"这个批处理调用"ISDDown.exe"软件将镜像文件写入 FLASH 中。这个是开源工具，有兴趣的可以自行查阅资料，来了解整个的流程。

2、通过 USB 插入电脑之后，点击"mydown.bat"，会弹出如下窗口

3、更新完成之后，会出现如下界面:

4、更新完成之后，就可以看到电脑已经虚拟出来了一个"CDROM"，到此就说明成功了。

备注：如果在更新过程中，出现任何异常，都是不正常的，可以更换 FLASH 来确定问题。

4.6 组合播放的参考例程

1、串口发送和校验

```
/*****************************************************************************************
 - 功能描述:  串口向外发送组合数据
 - 隶属模块:  外部
 - 参数说明:  CMD:表示控制指令，请查阅指令表，还包括查询的相关指令
              feedback:是否需要应答[0:不需要应答，1:需要应答]
              pbf:传送的参数的指针
              Len:传送的参数的长度
 - 返回说明:
 - 注:
*****************************************************************************************/
void Uart_SendDat(INT8U CMD ,INT8U feedback , INT8U *pbf ,INT8U Len)
{
    static u8 i = 0 ;

    Send_buf[0] = 0xff;     //保留字节
    Send_buf[1] = 4 + Len ;     //长度
    Send_buf[2] = CMD;      //控制指令
    Send_buf[3] = feedback;//是否需要反馈
    for(i=0 ; i<Len ; i++)
    {
        Send_buf[4+i] = *(pbf+i) ;
    }
    DoSum(&Send_buf[0],Send_buf[1]);//校验
    SendCmd(Send_buf[1]+2 , 1 , 30);//发送此帧数据
}
```

```
/*****************************************************************************
 - 功能描述：求和校验
 - 隶属模块：
 - 参数说明：
 - 返回说明：
 - 注：
*****************************************************************************/
void DoSum( INT8U *Str, INT8U len)
{
    INT16U xorsum = 0;
    INT8U i;

    for(i=0; i<len; i++)
    {
        xorsum  = xorsum + Str[i];
    }
    xorsum      = 0 -xorsum;
    *(Str+i)    = (INT8U)(xorsum >>8);
    *(Str+i+1)  = (INT8U)(xorsum & 0x00ff);
}
```

2、函数的调用关系：

u8 UartDatCnt ;//组合的参数长度，用户自己指定的数据长度

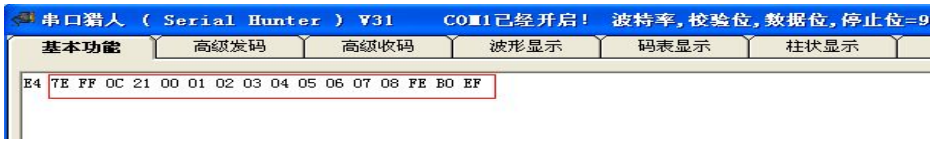
u8 PDatBuf[12] ;//定义数据的缓冲区

void DatChangeToBuf(u8 len)//这里单纯是赋值测试使用，用户自行改写代码

```
{
    static u8 i = 0 ;
    UartDatCnt = len ;
    for(i=0 ; i<len ; i++)
    {
        PDatBuf[i] = i+1 ;
    }
}
```

Uart_SendDat(0x21 ,0 , PDatBuf ,UartDatCnt) ;//发送组合播放的数据包

3、实际发送的数据效果



实际的数据：7E FF 0C 21 00 01 02 03 04 05 06 07 08 FE B0 EF

备注:还有不清晰的地方，可以联系我们。提供技术支持

4.7 模块的硬件说明

- 模块供电
- 模块的指示灯
- 3.5 音频接口

1、模块的供电

模块最理想的工作电压为 4.2V。所以用户如果是采用 5V 的电源供电的，建议串接一个二极管

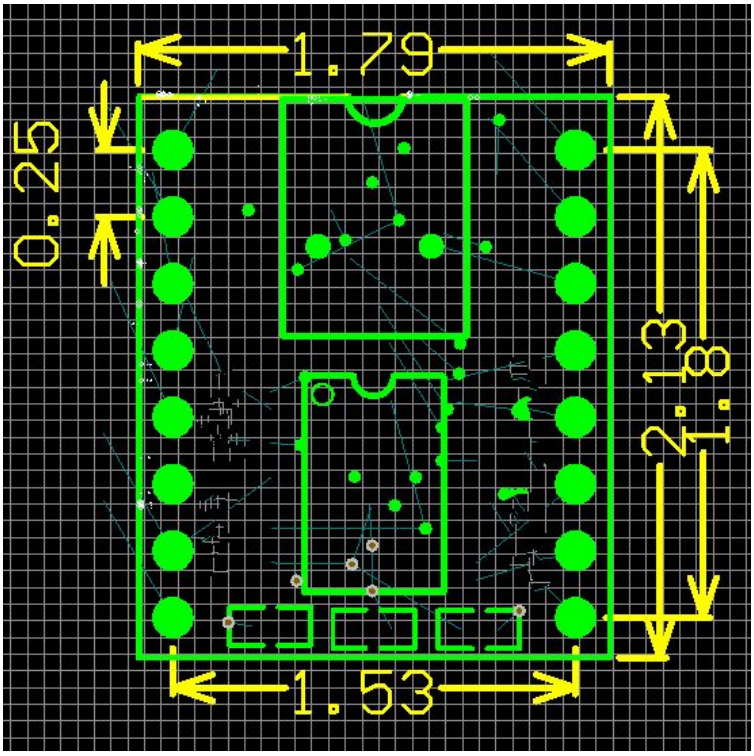
2、模块的指示灯

	下载模式	播放语音	暂停	睡眠
状态	快闪	慢闪	常亮	灭

3、3.5 音频接口

此接口是供用户插上自己的耳机来听语音的，只能支持耳机的，不能跳线出来连接喇叭的

4.8 模块尺寸



以上尺寸单位为 CM

未标注的均以 mm 为单位，排针的间距为2.54mm

5. 注意事项

模块的使用,关键的地方做如下说明:

- 模块的 GPIO 的特性
- 应用的中注意事项
- 串口编程部分的注意

5.1 GPIO 的特性

IO 输入特性						
符号	参数	最小	典型	最大	单位	测试条件
V_{IL}	Low-Level Input Voltage	-0.3	-	0.3*VDD	V	VDD=3.3V
V_{IH}	High-Level Input Voltage	0.7VDD	-	VDD+0.3	V	VDD=3.3V
IO 输出特性						
符号	参数	最小	典型	最大	单位	测试条件
V_{OL}	Low-Level Output Voltage	-	-	0.33	V	VDD=3.3V
V_{OH}	High-Level Output Voltage	2.7	-	-	V	VDD=3.3V

5.2 应用中的注意点

1、模块对外的接口均是 3.3V 的 TTL 电平，所以在硬件电路的设计中，请注意电平的转换问题。另外在强干扰的环境中，请注意电磁兼容的一些保护措施，GPIO 采用光耦隔离，增加 TVS 等等

2、ADKEY 的按键取值均按照一般的使用环境，如果在强感性或者容性负载的环境下，请注意模块的供电，建议采用单独的隔离供电，另外再配上磁珠和电感对电源的滤波，一定要尽可能的保证输入电源的稳定和干净。如果实在无法保证，请联系我们，减少按键的数量，重新定义更宽的电压分配。

■ 串口通信，在一般的使用环境下，注意好电平转换即可。如果强干扰环境，或者长距离的 RS485 应用，那么请注意信号的隔离，严格按照工业的标准设计通信电路。可以联系我们，我们提供设计参考

■ 我们支持音频文件的采样率最低为 8KHZ。也就是说低于 8KHZ 的音频文件是不支持的，不能正常解码播放。用户可以使用音频处理软件，提高音频文件的采样率来解决这个问题。

■ 模块在睡眠状态的电流在 10MA 左右，播放中，依据音量的大小，峰值电流可以达到 1A。功耗会比较大。如果使用在低功耗场合，请用户控制模块或者芯片的供电。这样可以减小芯片的功耗

■ 用户如果直接使用我们模块自带的功放，请选择合适的喇叭即可。推荐使用 4 欧姆/3W。这个是使用效果最好的配置。选用其它的喇叭，请注意负载大小，以及功率这两个参数

■ 该模块模块支持 MP3、WAV 二种主流的音频格式。

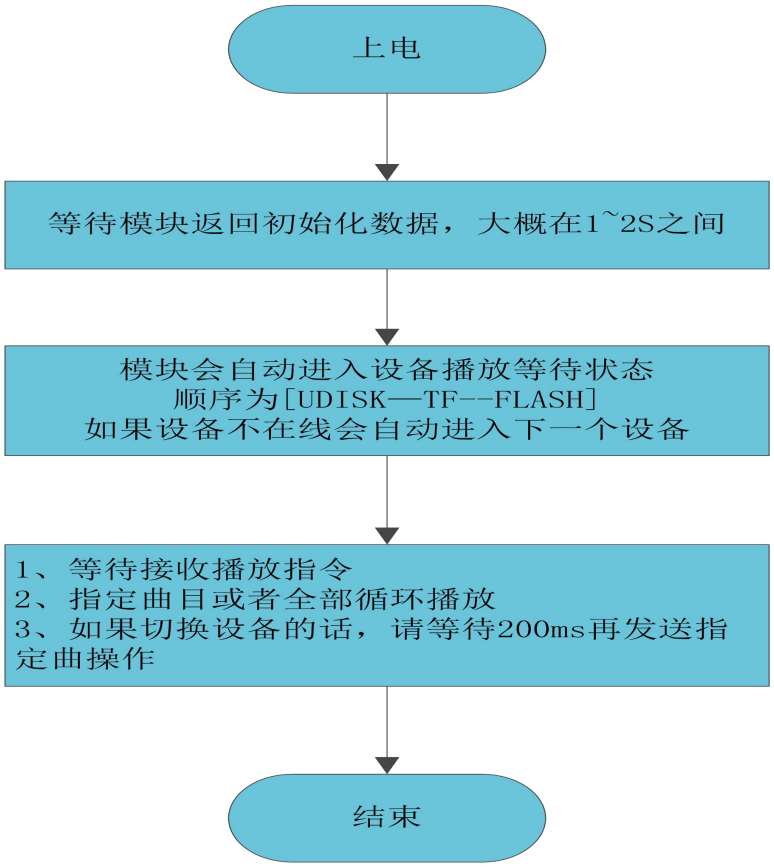
8、我们的模块支持 8/11.025/12/16/22.05/24/32/44.1/48KHZ 采样率的音频文件，这些也是网络上绝大多数的音频文件的参数。如果用户的音频文件的采样率不在此范围内，是不支持播放的，但是可以通过专用的软件转换一下即可。我们的优势就是无损播放和高音质，所以不太建议用户对音频进行压缩。

5.3 串口操作

串口部分的操作，参见下面的流程，我们提供了完整的参照例程，供用户参考:

1. 串口的操作流程
2. 串口编程参考的说明
3. 串口操作需要延时的注意事项

5.3.1 串口操作流程



1、我司提供的所有模块的串口部分的操作，均是一样的协议，所以不用担心不同模块的不兼容
2、如果对串口的操作，有任何不明白的，请一定联系我们，索取串口编程参考例程。
3、我们产品的更新，也一定会按照当前的协议版本，做到向下兼容。

5.3.2 串口编程参考的说明

目前我们提供的串口编程参考代码，有两部分，第一部分是我们测试版的测试代码，相关的串口操作比较全面，另一个是基本版，只是指定曲目的范例。请用户耐心消化

5.3.3 串口编程需要适当延时的注意点

1、模块上电之后，需要大概不到 500ms 的时间进行初花的相关操作，初始化完毕之后，会有初始化的相关数据发送出来。用户也可以直接不理会这些数据
2、当指定设备播放之后，需要延时 200ms 的时间，再发送指定曲目等等相关指令。
3、由于 SPIFLASH 本身能存储的曲目很有限，不像 TF 卡或者 U 盘。所以指定文件播放，到语音开始播放之间的延时，可以忽略不计。

5.3.4 校验的重要说明

1、争对很多用户不太习惯校验的通信方式，我们特别推出了带校验和不带校验的兼容方式。举例说明。如果我们发送组合播放指令如下:

组合播放[不带校验]	7E FF 09 21 00 05 01 02 03 04 EF	播放5、1、2、3、4
组合播放[带校验]	7E FF 09 21 00 05 01 02 03 04 FE C8 EF	播放5、1、2、3、4

比较两条指令的区别，就是省略掉的校验的 2 个字节。这两帧数据均可以被芯片所接收。
2、因为很多用户在使用的过程中，很多都是使用不带晶振的 MCU。这样的话，我们必须建议您加上校验这种方式，来保证通信的稳定性。
3、假如用户使用 STM32 或者 STC 等等 MCU，并且是外挂晶振的，就可以适当的省掉校验。因为不带晶振的 MCU，时钟是相对不那么准的，所以串口会存在误差，一旦误差过大，会导致通信出错。请用户朋友自行斟酌。

6. 免责声明

- 开发预备知识

MSK330 系列产品将提供尽可能全面的开发模版、驱动程序及其应用说明文档以方便用户使用但也需要用户熟悉自己设计产品所采用的硬件平台及相关 C 语言的知识

■ EMI 和 EMC

MSK330 系列模块机械结构决定了其 EMI 性能必然与一体化电路设计有所差异。MSK330 系列模块的 EMI 能满足绝大部分应用场合，用户如有特殊要求，必须事先与我们协商.

MSK330 系列模块的 EMC 性能与用户底板的设计密切相关，尤其是电源电路、I/O 隔离、复位电路，用户在设计底板时必须充分考虑以上因素。我们将努力完善 MSK330 系列模块的电磁兼容特性，但不对用户最终应用产品 EMC 性能提供任何保证.

■ 修改文档的权力

本公司能保留任何时候在不事先声明的情况下对产品相关文档的修改权力

■ ESD 静电放点保护

MSK330 系列产品部分元器件内置 ESD 保护电路，但在使用环境恶劣的场合，依然建议用户在设计底板时提供 ESD 保护措施，特别是电源与 IO 设计，以保证产品的稳定运行，安装 MSK330 系列产品为确保安全请先将积累在身体上的静电释放，例如佩戴可靠接地的静电环，触摸接入大地的自来水管等

版 本 历 史

版本	日期	原因
V1.0	2013/09/18	内测版本
V1.1	2013/12/10	正式发布版本
V1.2	2014/02/18	1、增加指定文件夹和文件地址播放[文件分区管理] 2、增加按键的多种模式可选 3、增加指定文件夹循环播放 4、查询更多的系统参数 5、增加文档的详细说明