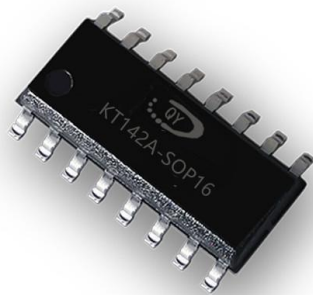


# KT142A4 芯片使用手册

文件状态：	文件标识：	
<input type="checkbox"/> 草稿	当前版本：	V2
<input checked="" type="checkbox"/> 正式发布	作 者：	
<input type="checkbox"/> 正在修改	完成日期：	2022-07-22



## 版本历史

版本	日期	原因
V1.0	2021/12/12	内测版本和优化版本
V1.1	2022/03/22	1、完善文档的描述性错误 2、新增多文件夹的指定播放，详见3.4.7
V1.2	2022/04/12	1、完善文档的描述性错误 2、新增低功耗的说明，详见3.4.5 3、新增 busy 脚的电平设置，新增低功耗模式设置。详见3.4.15 4、新增音频输出切换指令，详见3.4.16 5、新增版本查询的指令，详见3.5.5
V1.3	2022/04/14	1、新增参考指令举例说明，详见3.2.3章节 2、完善一些描述性错误
V2	2022/07/22	1、新增章节5，关于使用芯片内置空间的说明 2、完善文档的描述性错误 3、新增章节8的问题集锦

# 目 录

1. 概述 .....	5
1.1 简介 .....	5
KT142A 是一个提供串口的语音芯片，完美的集成了 MP3 的硬解码。内置 332KBYTE 的空间，最大支持 332 秒的语音长度，支持多段语音，内置空间不够也可以选择外挂 SPIFLASH 存储器，同时支持直驱 0.5W 的扬声器，无需功放 .....	
软件支持串口通信协议，可以外挂 SPIFLASH、U 盘、TF 卡作为存储介质，用户可以灵活的选用其中的任何一种设备作为语音的存储介质。通过简单的串口指令即可完成播放指定的语音，以及如何播放等功能，无需繁琐的底层操作，使用方便，外围简单，稳定可靠是此款产品的最大特点。 .....	
1.2 功能 .....	5
2. 方案说明 .....	6
2.1 参数说明 .....	6
2.2 管脚说明 .....	7
3. 串口通讯协议 .....	8
3.1 通讯格式 .....	8
3.2 通讯指令 .....	9
3.2.1 控制指令 .....	9
3.2.2 查询指令 .....	10
3.2.3 通讯指令举例 .....	11
3.3 芯片返回的数据 .....	12
3.3.1 芯片上电返回的数据[3F] .....	12
3.3.2 曲目播放完毕返回的数据[3C][3D][3E] .....	12
3.3.3 模块应答 ACK 返回的数据[41] .....	13
3.3.4 模块错误返回的数据[40] .....	13
3.3.5 设备插入拔出消息[3A][3B] .....	13
3.4 串口控制指令详解 .....	14
3.4.1 指定歌曲播放指令[0x03] .....	14
3.4.2 指定音量播放指令[0x06] .....	14
3.4.3 单曲循环播放指令[0x08] .....	14
3.4.4 指定播放设备[0x09] .....	15
3.4.5 指定波特率和复位指令和低功耗设置[0x0B][0x0C][0x0A] .....	15
3.4.6 指定文件夹文件名播放[0x0F] .....	16
3.4.7 指定 9999 文件夹文件名播放[0x10] .....	16
3.4.8 全部循环播放指令[0x11] .....	17
3.4.9 播放停止指令[0x15][0x16] .....	17
3.4.10 指定文件夹开始循环顺序播放[0x17] .....	17
3.4.11 随机播放设备文件[0x18] .....	17
3.4.12 对当前的曲目设置为循环播放[0x19] .....	18

3.4.13 多文件夹插播功能[0x25]	18
3.4.14 指定文件夹循环随机播放[0x28]	19
3.4.15 恢复出厂设置和 BUSY 的电平设置[0xC1][0xC2]	19
3.4.16 设置芯片为 DAC 输出还是直驱扬声器[0x1A]	19
3.4.17 组合播放指令[0x21]	20
3.5 串口查询指令详解	21
3.5.1 查询当前在线的设备[0x3F]	21
3.5.2 播放状态查询指令[0x42]	21
3.5.3 指定文件夹曲目总数查询[0x4E]	22
3.5.4 当前设备的总文件夹数目查询[0x4F]	22
3.5.5 查询当前芯片的软件版本[0x46]	22
<b>4. 注意事项</b>	<b>23</b>
4.1 GPIO 的特性	23
详细的可以参加芯片的规格书	23
4.2 应用中的注意点	23
4.3 USB 更新语音说明	24
4.4 USB 更新 SPIFLASH 的语音详细说明	25
4.5 用户使用空白的 SPIFLASH 说明	26
4.6 注意事项点	27
4.6.1 芯片上电的工作流程图	27
4.6.2 串口编程需要适当延时的注意点	28
<b>5. 使用芯片内置空间说明</b>	<b>29</b>
5.1 MP3 文件特性介绍	29
5.2 第 1 步--产生音频文件	29
5.3 第 2 步--串口工具下载音频	30
5.4 内置空间的细节说明	31
<b>6. 免责声明</b>	<b>31</b>
<b>7. 订货信息</b>	<b>32</b>
7.1 参考原理图	32
7.2 封装尺寸	33
<b>8. 常见问题集锦</b>	<b>34</b>

## 1. 概述

### 1.1 简介

KT142A 是一个提供串口的语音芯片，完美的集成了 MP3 的硬解码。内置 332KByte 的空间，最大支持 332 秒的语音长度，支持多段语音，内置空间不够也可以选择外挂 spiflash 存储器，同时支持直驱 0.5W 的扬声器，无需外置功放

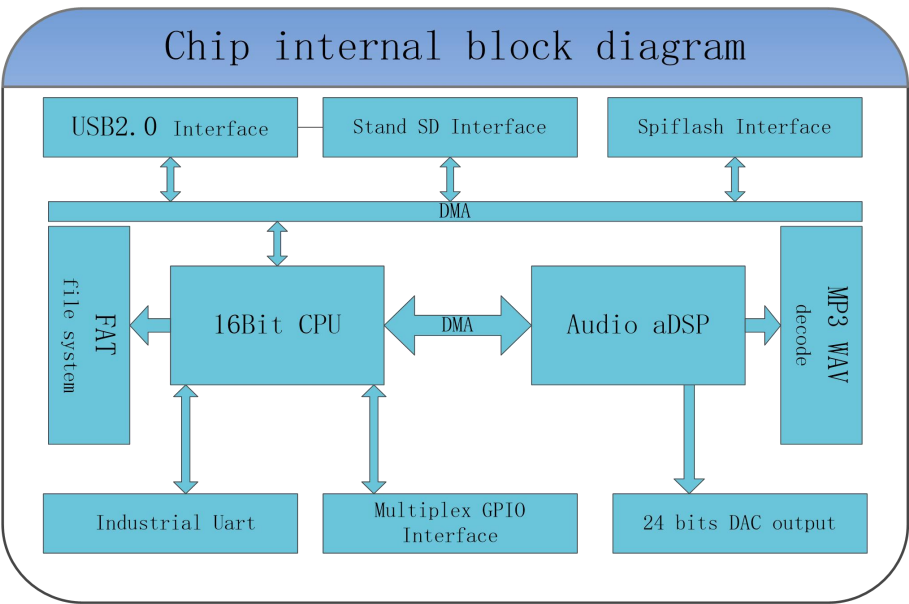
软件支持串口通信协议，可以外挂 SPIFLASH、U 盘、TF 卡作为存储介质，用户可以灵活的选用其中的任何一种或多种设备作为语音的存储介质。通过简单的串口指令即可完成播放指定的语音，以及如何播放等功能，无需繁琐的底层操作，使用方便，外围简单，稳定可靠是此款产品的最大特点。

1、支持使用我们配套的上位机软件，直接串口下载 mp3 文件到芯片内置 332kByte 存储空间中  
2、无需任何烧录器，无需任何软件，PC 虚拟外挂的 SPIFLASH 为存储介质的 U 盘功能, 章节 4.3  
3、芯片内置 0.5W 的功放，超宽的供电范围【2.5--5.2V】，外置 2 个 105 的电容即可正常工作，非常简单

### 1.2 功能

1、 <b>芯片支持重复烧写</b> ，支持 MP3 文件的采样率(KHz):8/11.025/12/16/22.05/24/32
2、24 位 DAC 输出，动态范围支持 90dB，信噪比支持 85dB 。 <b>芯片支持任意切换 DAC 输出和 PWM 直驱扬声器</b>
3、 <b>内置 300 多秒的语音空间</b> ，最大支持外挂 16M 字节的 SPIFLASH。例如 W25Q16[2M 字节]、W25Q128[16M 字节]
4、USB 接口更新语音文件，无需安装任何软件。支持 XP 和 WIN7、WIN10 等等系统。
5、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
6、支持外挂多种类型功放，如 3W、5W、10W 等等 。也支持内置 0.5W 的 PWM 直接驱动扬声器
7、支持各种功能的订制开发，使用非常的灵活多变
8、可以支持多种存储介质，如：U 盘、TF 卡、SPIFLASH[超小型封装]作为存储介质

2. 方案说明

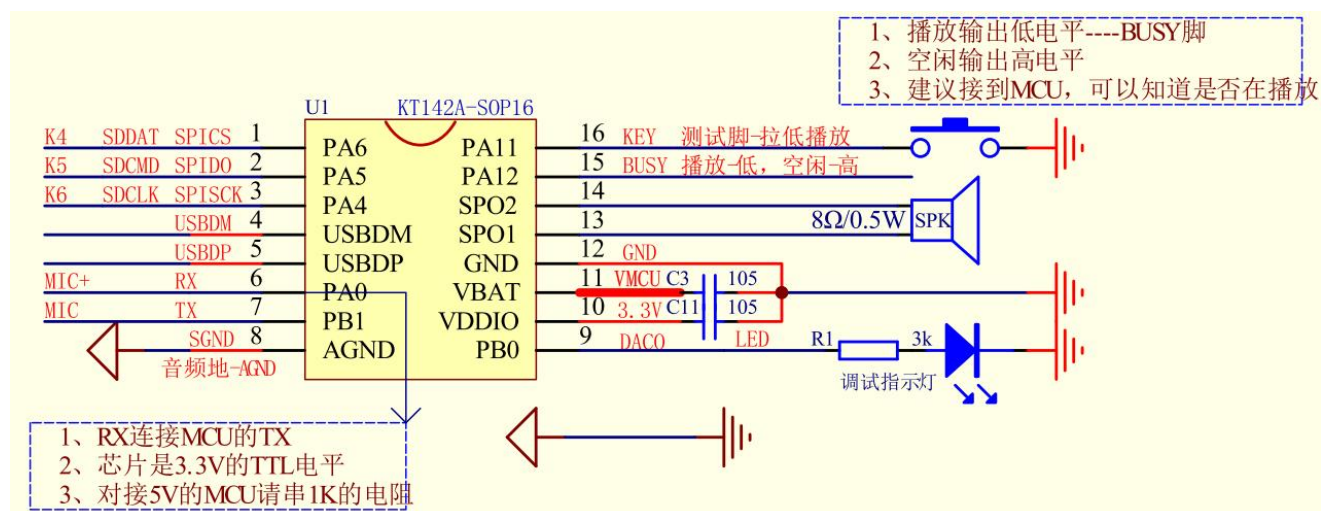


芯片是 SOC 方案，集成了一个 32 位的 RISC MCU，以及一个专门针对音频解码的 aDSP，采用硬解码的方式，更加保证了系统的稳定性和音质。小巧的封装尺寸更加满足嵌入其它产品的需求

2.1 参数说明

名称	参数
MP3文件格式	1、支持所有比特率11172-3和 IS013813-3 layer3音频解码
	2、采样率支持 (KHZ) :8/11.025/12/16/22.05/24/32
	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[指令设置]
输入电压	2.5V-5.2V
额定电流	10MA[静态]
低功耗电流	<50uA
功放功率	内置0.5W 的 PWM 输出
尺寸	参见封装章节
工作温度	[-40度] -- [80度]
湿度	5% ~ 95%
主芯片型号	KT142A[SOP16]

## 2.2 管脚说明



引脚序号	引脚名称	功能描述	备注
1	PA6	SPI 的片选	
2	PA5	SPI 的数据口	
3	PA4	SPI 的时钟口数字地	
4	USBDM	USB 接口/通用输入输出口	必须预留测试点
5	UDBDP	USB 接口/通用输入输出口	必须预留测试点
6	PA0	串口-RX	连接 MCU 的 TX 脚
7	PB1	串口-TX	连接 MCU 的 RX 脚
8	AGND	模拟音频地	
9	PB0	音频 DAC/指示灯脚	必须预留测试点 芯片音频 DAC 输出外加功放
10	VDDIO	3.3V LDO 输出[KT142A 输出]	必须预留测试点
11	VBAT	供电脚正极	4.2V 为最佳
12	GND	数字地	
13	SPO1	扬声器 PWM 输出脚 1	直驱 0.5W 扬声器[不分正负]
14	SPO2	扬声器 PWM 输出脚 1	直驱 0.5W 扬声器[不分正负]
15	PA12	通用输入输出口	有声音输出低，空闲输出高
16	PA11	通用输入输出口	测试脚

### 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 115200**

#### 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: 115200 bps --- 可以发送指令修改, 并且记忆, 详见 3.4.5				
数据位 : 8    停止位 : 1    校验位 : none    流控制 : none				
格式: \$S VER Len CMD Feedback para1 para2 checksum \$0				
1	0x7E		起始标记	
2	CMD		命令字节	
3	lenH		包的序号--高字节	
4	lenL		包的序号--低字节	
.....	DAT		数据区	
结束	0xEF		结束标志	

1、假如指定播放内置 flash 的里面的语音播放，就需要发送: 7E 03 00 02 00 01 EF  
==> 其中 0x03 代表的是命令字节，其中 [00 02] 是数据长度 2。其中 [00 01] 代表指定第 1 段语音播放

2、假如指定播放内置 flash 的里面的语音播放，就需要发送: 7E 03 00 02 00 64 EF  
==> 其中 0x03 代表的是命令字节，其中 [00 02] 是数据长度 2。其中 [00 64] 代表指定第 100 段语音播放



## 3.2 通讯指令

我们的通讯分为以下两大块

- 控制指令 -- 详见 3.2.1
- 查询芯片的参数以及状态--详见 3.2.2

### 3.2.1 控制指令

CMD 命令	对应的功能	参数(16位)
0x01	下一曲	
0x02	上一曲	
0x03	指定曲目(NUM)	详见3.4.1
0x04	音量+	
0x05	音量-	
0x06	指定音量	详见3.4.2
0x08	单曲循环指定曲目播放	详见3.4.3
0x09	指定播放设备	保留
0x0A	芯片进入低功耗状态	详见3.4.5
0x0B	指定波特率	详见3.4.5
0x0C	芯片复位	详见3.4.5
0x0D	播放	
0x0E	暂停	
0x0F	指定文件夹文件名播放	详见3.4.6
0x10	指定文件夹文件名播放，扩展9999个文件夹	详见3.4.7
0x11	指定当前的设备全部循环播放	
0x16	停止	
0x17	指定文件夹循环播放	详见3.4.10
0x18	指定当前的设备全部随机播放	详见3.4.11
0x19	对当前播放的曲目设置为循环播放	详见3.4.12
0x1A	芯片 dac 输出和直驱扬声器切换	详见3.4.16
0x21	组合播放	详见3.4.17
0x28	指定文件夹内部循环播放	详见3.4.14
0xC1	恢复出厂设置	详见3.4.15
0xC2	设置 busy 脚的电平	详见3.4.15

### 3.2.2 查询指令

这里是查询芯片的状态和相关的参数

CMD 命令详解(查询)	对应的功能	参数(16位)
0x3F	查询在线的设备	详见3.5.1
0x42	查询当前状态	详见3.5.2
0x43	查询当前音量	7E 43 00 02 00 00 EF
0x47	查询 U 盘总文件数	详见3.5.3
0x48	查询 TF 卡总文件数	详见3.5.3
0x49	查询外挂的 spiflash 总文件数	详见3.5.3
0x4A	查询内置的 spiflash 总文件数	详见3.5.3
0x4B	查询 U 盘当前的文件	详见3.5.4
0x4C	查询 TF 卡当前的文件	详见3.5.4
0x4D	查询外挂的 spiflash 当前的文件	详见3.5.4
0x4E	查询内置的 spiflash 当前的文件	详见3.5.4
0x62	查询当前设备的总文件数	详见3.5.4

### 3.2.3 通讯指令举例

命令类型	CMD 命令	参考指令	指令说明
控制指令	0x01	7E 01 00 02 00 00 EF	下一曲
	0x02	7E 02 00 02 00 00 EF	上一曲
	0x03	7E 03 00 02 00 64 EF	指定曲目(NUM) -- 播放第100段
	0x04	7E 04 00 02 00 00 EF	音量+
	0x05	7E 05 00 02 00 00 EF	音量-
	0x06	7E 06 00 02 00 0A EF	指定音量 -- 取值范围[0--30] --指定为10级
	0x08	7E 08 00 02 00 01 EF	单曲循环指定曲目播放 -- 循环播放第1段
	0x09	7E 09 00 02 00 01 EF	指定播放设备 U 盘--[02=TF][04=外挂 flash][08=内置]
	0x0A	7E 0A 00 02 00 01 EF	芯片进入低功耗状态
	0x0B	7E 0B 00 02 00 01 EF	指定波特率--1200
	0x0C	7E 0C 00 02 00 00 EF	芯片复位
	0x0D	7E 0D 00 02 00 00 EF	播放
	0x0E	7E 0E 00 02 00 00 EF	暂停
	0x0F	7E 0F 00 02 01 01 EF	指定文件夹文件名播放--01文件夹001号文件
	0x10	7E 10 00 03 04 D2 01 EF	<b>文件夹1234的001xxx.mp3</b>
	0x11	7E 11 00 02 00 01 EF	指定当前的设备全部循环播放
	0x16	7E 16 00 02 00 00 EF	停止
	0x17	7E 17 00 02 00 01 EF	指定文件夹循环播放 -- 指定01文件夹循环播放
	0x18	7E 18 00 02 00 01 EF	指定当前的设备全部随机播放
	0x19	7E 19 00 02 00 00 EF	对当前播放的曲目设置为单曲循环【播放状态发送有效】
	0x1A	7E 1A 00 02 00 01 EF	芯片 dac 输出和直驱扬声器切换 -- 设置为扬声器输出
	0x21	7E 21 00 06 01 01 01 03 02 02 EF	组合播放-[1,1][1,3][2,2] , <b>内置 flash 详见3.4.17</b>
	0x28	7E 28 00 02 00 01 EF	指定文件夹内部随机播放 -- 01文件夹
	0xC1	7E C1 00 02 00 00 EF	恢复出厂设置
	0xC2	7E C2 00 02 00 01 EF	设置 busy 脚的电平--空闲输出低, 播放输出高
查询指令	0x3F	7E 3F 00 02 00 00 EF	查询在线的设备
	0x42	7E 42 00 02 00 00 EF	查询当前状态
	0x43	7E 43 00 02 00 00 EF	查询当前音量
	0x46	7E 46 00 02 00 00 EF	查询当前的版本
	0x47	7E 47 00 02 00 00 EF	查询 U 盘总文件数
	0x48	7E 48 00 02 00 00 EF	查询 TF 卡总文件数
	0x49	7E 49 00 02 00 00 EF	查询外挂的 spiflash 总文件数
	0x4A	7E 4A 00 02 00 00 EF	查询内置的 spiflash 总文件数
	0x4B	7E 4B 00 02 00 00 EF	查询 U 盘当前的文件
	0x4C	7E 4C 00 02 00 00 EF	查询 TF 卡当前的文件
	0x4D	7E 4D 00 02 00 00 EF	查询外挂的 spiflash 当前的文件
	0x4E	7E 4E 00 02 00 00 EF	查询内置的 spiflash 当前的文件
	0x62	7E 62 00 02 00 00 EF	查询当前设备的总文件数

### 3.3 芯片返回的数据

芯片在关键地方均会有数据返回。供用户掌控芯片的工作状态

- 芯片上电初始化成功的数据
- 芯片播放完当前曲目的数据
- 芯片成功接收到指令返回的 ACK(应答)
- 芯片接收一帧数据出错[包括数据没收完整、校验出错两种情况]
- 芯片在繁忙时，有数据过来，芯片会返回忙的指令
- U 盘、TF 卡插入拔出，均有数据返回

#### 3.3.1 芯片上电返回的数据[3F]

U 盘和 TF 卡都在线，其他都不在线	7E 3F 00 02 00 03 EF
只有外挂 FLASH 在线	7E 3F 00 02 00 08 EF

数据和设备对应的关系---每个设备都是或的关系	
0x01=BIT(0)	当前设备是 U 盘
0x02=BIT(1)	当前设备是 TF 卡
0x04=BIT(2)	当前设备连接电脑
0x08=BIT(3)	当前设备是外置 flash 模式
0x10=BIT(4)	当前是内置 flash 模式

1、芯片上电，需要一定的时间初始化，这个时间是需要根据设备的文件多少决定的，一般在 700 毫秒。如果超过这个时间芯片的初始化数据还没有发送出来，说明芯片初始化出错，请检查硬件

2、芯片初始化返回的数据为当前的有效文件夹，譬如返回 7E 3F 02 00 03 EF

==>其中 0x03 代表的是 U 盘和 TF 这两个设备在线

3、MCU 必须等待芯片初始化指令发出之后才能发送相应的控制指令，否则发送的指令芯片将不予处理。

#### 3.3.2 曲目播放完毕返回的数据[3C][3D][3E]

U 盘播放完第1曲	7E 3C 00 02 00 01 EF
TF 卡播放完第2曲	7E 3D 00 02 00 02 EF
外挂 FLASH 播放完第1曲	7E 3E 00 02 00 01 EF

1、芯片默认播放完毕之后，自动停止，等待用户发送，busy 脚作为播放和未播放的状态指示。**请参见第 BUSY 脚**

2、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，因为一旦指定设备之后，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、设备播放完每一段语音，都会有数据返回，举例说明:7E 3C 00 02 00 01 EF

(1)、其中 0x3C 代表的是 U 盘，详细的请参见上表

(2)、其中 0x00,0x01 代表的是第 1 段语音播放完毕。0x01,0xF4=第 500 段声音。0x01F4 = 500. 这里代表的是高低字节

4、由于设备中所有的文件均是以物理顺序进行识别的，包括指定文件夹文件名播放，播放完，返回的还是物理顺序编号

### 3.3.3 模块应答 ACK 返回的数据[41]

7E 41 EF	芯片主动返回，说明成功接收到数据
----------	------------------

为了加强数据通信之间的稳定性，我们增加应答处理。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。

### 3.3.4 模块错误返回的数据[40]

返回忙	7E 40 00 02 00 01 EF	芯片在文件系统初始化时
当前是睡眠模式	7E 40 00 02 00 02 EF	睡眠模式只支持指定设备
串口接收错误	7E 40 00 02 00 03 EF	串口一帧数据没接收完毕
校验出错	7E 40 00 02 00 04 EF	和校验出错
指定文件超范围	7E 40 00 02 00 05 EF	文件的指定超过设定的范围
未找到指定文件	7E 40 00 02 00 06 EF	指定为文件没有被找到
插播错误	7E 40 00 02 00 07 EF	插播只允许在播放的状态下进行
播放 TF 卡错误	7E 40 00 02 00 08 EF	TF 卡读取失败或者 TF 卡被拔出或者有坏区
FLASH 初始化出错	7E 40 00 02 00 09 EF	FLASH 里面的文件系统信息错误
进入睡眠	7E 40 00 02 00 0A EF	进入 SLEPP 模式

- 1、为了加强数据通信之间的稳定性，增加了数据错误处理机制。模块收到不符合格式的数据，或者异常均会有信息反馈
- 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、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。

### 3.3.5 设备插入拔出消息[3A][3B]

U 盘插入	7E 3A 00 02 00 01 EF
U 盘拔出	7E 3B 00 02 00 01 EF
TF 插入	7E 3A 00 02 00 02 EF
TF 拔出	7E 3B 00 02 00 02 EF
PC 插入	7E 3A 00 02 00 04 EF
PC 拔出	7E 3B 00 02 00 04 EF

- 1、为了加强模块灵活性，我们特别增加了，设备插入、拔出的指令反馈。方便知道模块的工作状态。
- 2、设备插入的时候，我们默认进入到设备等待状态，如果用户插入的是带灯的 U 盘，可以看到 U 盘灯闪烁。也可以接收到设备插入的串口消息。

### 3.4 串口控制指令详解

以下我们对关键的地方进行详细的说明--面向控制指令：

#### 3.4.1 指定歌曲播放指令[0x03]

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 0~3000. 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。此指令在 TF 卡和 U 盘状态是按照存储的物理顺序指定的。FLASH 同样也是按照物理顺序[拷贝的先后顺序]

数据	详解--例如选择第 2 首歌播放，串口的发送部分 7E 03 00 02 00 02 EF
0x7E	起始字节
0x03	命令字节
0x00	长度--高字节
0x02	长度--低字节
0x00	曲目的高字节 [DH]
0x02	曲目的高字节 [DL]
0xEF	结束字节

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

#### 3.4.2 指定音量播放指令[0x06]

指定音量为30级--不记忆	7E 06 00 02 00 1E EF
指定音量为10级--带记忆	7E 08 00 02 01 0A EF

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, 取值范围 0--30
2、DH = 0x00 ； DL = 0x1E ， 转化为 16 进制为 0x001E。音量为 30 级，这个是不带记忆的 ==> 芯片下次上电或者复位还是最大，也就是 30 级
1、DH = 0x01 ； DL = 0x0A ， 转化为 16 进制为 0x011A。音量为 10 级，注意这个是带记忆的 ==> 这里的 DH 设置为 1 之后，芯片就会自动进入记忆音量，设置音量为 10 级之后，芯片复位或者下次上电则继续是 10 级 。 如果不需要记忆音量，则将这里的 DH 改成 0 发送即可

#### 3.4.3 单曲循环播放指令[0x08]

循环播放第一曲	7E 08 00 02 00 01 EF
循环播放第二曲	7E 08 00 02 00 02 EF

- 1、面向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。在操作 TF 卡或者 U 盘/以及 FLASH 时，按照的是文件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，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停止来关闭循环播放状态

### 3.4.4 指定播放设备[0x09]

指定播放设备-U 盘	7E 09 00 02 00 01 EF
指定播放设备-TF 卡	7E 09 00 02 00 02 EF
指定播放设备-外挂 spiflash	7E 09 00 02 00 04 EF
指定播放设备-内置的 spiflash	7E 09 00 02 00 08 EF
指定播放设备-PC	7E 09 00 02 00 10 EF

1、我们的模块默认是支持 4 种类型的播放设备, 只有设备在线才能指定设备去播放。设备是否在线, 我们软件会自动检测, 无需用户关系。如上表, 选择合适的指令发送。

2、指定设备之后。模块会自动进入停止解码状态, 等待用户指定曲目播放。从接收到指定设备到模块内部完成初始化文件系统。大概需要 200ms。请等待 200ms 之后再发送指定曲目的指令。

### 3.4.5 指定波特率和复位指令和低功耗设置[0x0B][0x0C][0x0A]

设置波特率600	7E 0B 00 02 00 00 EF	设置波特率19200	7E 0B 00 02 00 05 EF
设置为1200	7E 0B 00 02 00 01 EF	设置38400	7E 0B 00 02 00 06 EF
设置为2400	7E 0B 00 02 00 02 EF	设置57600	7E 0B 00 02 00 07 EF
设置为4800	7E 0B 00 02 00 03 EF	设置115200	7E 0B 00 02 00 08 EF
设置为9600	7E 0B 00 02 00 04 EF	设置256000	7E 0B 00 02 00 09 EF
复位	7E 0C 00 02 00 00 EF		
进入低功耗	7E 0A 00 02 00 01 EF		

1、一旦设置了波特率之后, 芯片会记忆。下一次开机, 波特率就变成了您所设置的. 不支持查询
2、设置完波特率之后, 请等待 1 秒钟, 再发送复位[0x0C 指令], 或者断电一下即可
3、如果要恢复默认的波特率, 请发送恢复出厂设置的命令[0xC1], 此时芯片会自动擦除所有的配置
4、7E 0C 00 02 00 00 EF 这条是复位指令, 任何状态下面发送都有效。芯片内部时钟误差低于 1%, 所以波特率调高了, 也没问题

这里重点描述一下 7E 0A 00 02 00 01 EF 低功耗的指令
<p>1、芯片上电之后, 是正常工作模式, 也就是所有外设都是正常状态, 不播放保持在 6mA 的待机电流, 不同的设备电流消耗不一样。如果用户需要低功耗的场景, 就有两种解决方案</p> <p>2、方法 1: 外加一个 mos 管来控制芯片的电源, 当不需要芯片工作的时候主动切断芯片的供电, 需要的时候再供电, 芯片从掉电到上电正常接收指令, 大概需要 700ms</p> <p>3、方法 2: 可以给芯片发送 7E 0A 00 02 00 01 EF 这条指令, 芯片会进入待机模式, 功耗在 30uA 的样子。如果需要芯片再次工作起来, 可以串口发送任意一条指令, 然后等待 30ms, 芯片就正常起来了</p> <p>==》芯片的唤醒的原理是: 下降沿唤醒, 所以发送任意串口指令都可以唤醒芯片, 但是芯片没办法识别指令所以, 这里用户在低功耗状态下, 流程是: 先发任意指令唤醒---等待 30ms---再发需要的指令</p> <p>==》芯片被唤醒之后, 必须在 5 秒之内接收一次正确的串口数据帧, 否则会再次自动进入低功耗, 目的是防止误触发唤醒</p>

3.4.6 指定文件夹文件名播放[0x0F]

文件夹01的001xxx.mp3	7E 0F 00 02 01 01 EF
文件夹11的100xxx.mp3	7E 0F 00 02 0B 64 EF
文件夹99的255xxx.mp3	7E 0F 00 02 63 FF EF

- 1、默认文件夹的命名方式为“01”,“11”这样的方式，为了系统的稳定性和歌曲切换的速度，每个文件夹下默认最大支持 255 首歌,最多支持 99 个文件夹
- 2、例如指定“01”文件夹的 100xxx.MP3 文件,串口发送的指令为:7E 0F 02 01 64 EF
- DH:代表的是文件夹的名字,默认支持 99 个文件,即 01 -- 99 的命名
- DL:代表的是曲目,默认最多 255 首歌，即 0x01 ~ 0xFF
- 3、指定文件夹和指定曲目是支持 MP3、WAV 。当然文件夹和文件名都支持后面加汉字或者字符，规则如下：

脑 > data (E:) > 音频测试 > 29\_KT142A文件夹测试 >

名称	修改日期	类型
01	2022-03-21 17:51	文件夹
02数字10-19	2022-03-21 17:51	文件夹
03数字20-29	2022-03-21 17:51	文件夹
04数字30-39	2022-03-21 17:51	文件夹
05你好你好05	2022-03-21 18:02	文件夹

文件夹的命名，支持单独的数字，也支持数字+汉字的组合，但是不能超过 12 个字节  
==> 汉字占 2 个字节，数字和字母占 1 个字节

脑 > data (E:) > 音频测试 > 文件夹测试 > 05你好你好05

名称	#	标题	参与创作的艺术家
001回家.mp3			
002不得不爱.mp3		弦子&潘...	
003简单爱.mp3			
004越狱.mp3			

文件名的命名，也支持单独的数字，如“001.MP3”或者“001 回家.MP3”  
所支持的长度，和文件夹一样，不超过 12 个字节

3.4.7 指定 9999 文件夹文件名播放[0x10]

文件夹1234的001xxx.mp3	7E 10 00 03 04 D2 01 EF
文件夹8888的005xxx.mp3	7E 10 00 03 22 B8 05 EF

- 1、原理和 0x0F 指令是一样的，参见 3.4.6 章节
- 2、例如指定“1234\*”文件夹的 001xxx.MP3 文件,串口发送的指令为:7E 10 00 03 04 D2 01 EF
- DH= 04 D2 [两个字节组成] = 0x04D2 = 1234[转换为 10 进制] :代表的是文件夹的名字,即 1234
- DL= 01:代表的是曲目,默认最多 255 首歌，即 0x01 ~ 0xFF
- 3、指定文件夹和指定曲目是支持 MP3、WAV 。当然文件夹和文件名都支持后面加汉字或者字符，规则如下：

<p>1234多文件夹      2022-03-21 17:52      文件夹</p> <p>8888多文件夹      2022-03-21 17:52      文件夹</p> <p>文件夹的命名，支持单独的数字，也支持数字+汉字的组合，但是不能超过 12 个字节</p> <p>==》汉字占 2 个字节，数字和字母占 1 个字节</p>	<p>001蔡枫叛逆.mp3      叛逆</p> <p>002不将就.mp3</p> <p>003故事的角.mp3      故事的角色</p> <p>004曾经的你.mp3</p> <p>文件名的命名，也支持单独的数字，如“001.MP3”或者“001 回家.MP3”</p> <p>所支持的长度，和文件夹一样，不超过 12 个字节</p>
--	---



### 3.4.8 全部循环播放指令[0x11]

循环播放开始	7E 11 00 02 00 01 EF
循环播放停止	7E 11 00 02 00 00 EF

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

### 3.4.9 播放停止指令[0x15][0x16]

停止播放插播广告，回到背景音乐继续播	7E 15 00 02 00 00 EF
停止软件解码	7E 16 00 02 00 00 EF

- 1、在模块的播放过程中，我们有两种停止方式，一种是停止当前的插播广告，回到当前断点处继续播放背景音乐。另一种是停止所有的播放，包括背景音乐
- 2、假如当前在播放插播广告，这时发送停止指令 0x16，芯片会停止所有播放任务.0x16 的停止指令的优先级别是最高的，请用户留意

### 3.4.10 指定文件夹开始循环顺序播放[0x17]

指定02文件夹循环播放	7E 17 00 02 00 02 EF
指定01文件夹循环播放	7E 17 00 02 00 01 EF

- 1、对于 TF 卡和 U 盘或者 SPIFLASH，文件夹的命名方式必须是” 01” --- “99”。不可以超过 99
- 2、一旦指定文件夹循环之后，可以使用播放/暂停/上一曲/下一曲。这些操作命令都不会打断当前的文件夹循环播放状态。也就是说，发送下一曲指令之后，还是会循环当前的文件夹。
- 3、用户可以发送停止指令 0x16 来结束循环播放，返回至触发播放状态

### 3.4.11 随机播放设备文件[0x18]

整个设备的随机播放	7E 18 00 02 00 00 EF
-----------	----------------------

- 1、此指令是随机播放设备里面存储的所有语音文件，是按照物理顺序随机播放，不分设备里面是否带有文件夹。并且播放的第一个语音文件一定是设备里面的第一个语音文件

3.4.12 对当前的曲目设置为循环播放[0x19]

单曲循环播放开启	7E 19 00 02 00 00 EF
单曲循环播放关闭	7E 19 00 02 00 01 EF

- 1、需要在播放的过程中发送此指令，会循环播放当前的曲目。如果当前是处理暂停或者停止状态，则芯片不会响应此指令
- 2、如果要关闭单曲循环播放，发送关闭的指令即可，这样会把当前的曲目播放完毕之后，就停止。

3.4.13 多文件夹插播功能[0x25]

插播“ADVERT1”的文件夹，曲目为“001”	7E 25 00 02 01 01 EF
插播“ADVERT1”的文件夹，曲目为“002”	7E 25 00 02 01 02 EF
插播“ADVERT2”的文件夹，曲目为“001”	7E 25 00 02 02 01 EF




- 1、在之前我们的插播基础上，我们增加了多文件夹的插播功能，命名的方式如上图
- 2、文件夹最多支持 9 个，也就是从 ADVERT1 --- ADVERT9，文件夹的命令请一定按照我们给出的规则，否则会导致出错。请注意上图的文件夹命名格式
- 3、单个插播文件夹下的文件最大不能超过 255，也就是“255xxx.MP3/WAV”，请注意上图的文件名的命名格式。
- 4、完善了插播的相关应用，如当前播放的文件处于单曲循环或者当前文件夹循环，即使有插播进来，也不会改变当前状态，还会是单曲循环或者文件夹循环播放，除非用户使用停止指令。
- 5、插播的原理请参见 3.4.6。
- 6、此插播的功能，支持 TF 卡、U 盘以及 SPIFLASH。插播的文件仅仅允许在同一个设备内进行。
- 7、在停止状态，可以直接播放 ADVERTn 文件夹里面的曲目，就像指定文件夹文件名播放一样。
- 只要是播放 ADVERTn 文件夹里面的曲目，就不存在插播的问题，也就是说，当前播放的“ADVERT1”里面的 001 文件，在没有播放完之前，还可以继续指定播放 002 文件[此时会打断 001 文件]。

3. 4. 14 指定文件夹循环随机播放[0x28]

指定01文件夹随机播放	7E 28 00 00 01 EF
指定02文件夹随机播放	7E 28 00 00 02 EF
指定03文件夹随机播放	7E 28 00 00 03 EF

1、指定文件夹随机播放是我们制定的扩展功能，默认文件夹的命名方式为“01”，“11”这样的方式，为了系统的稳定性和歌曲切换的速度，每个文件夹下默认最大支持 255 首歌，最多支持 99 个文件夹

2、SPIFLASH 的操作和 TF 卡以及 U 盘一致，只是 SPIFLASH 的空间有限，能存放的语音数量有限



截图说明文件夹的命名方式

3. 4. 15 恢复出厂设置和 busy 的电平设置[0xC1][0xC2]

恢复出厂设置	7E C1 00 02 00 00 EF
设置 busy 脚，空闲输出低，播放输出高	7E C2 00 02 00 01 EF
设置 busy 脚，空闲输出高，播放输出低	7E C2 00 02 00 00 EF

1、恢复出厂设置这条指令，其实就是将之前设置的参数恢复到默认的值

(1)、波特率恢复到 9600，

(2)、busy 脚恢复到默认：空闲输出高电平，播放输出低电平

(3)、工作模式为普通模式，也就是非低功耗的状态

2、设置 busy 脚的状态指令，注意这条指令是带记忆的，发送设置之后，芯片会自动保存，下次上电有效

3. 4. 16 设置芯片为 DAC 输出还是直驱扬声器[0x1A]

设置芯片为直驱扬声器	7E 1A 00 02 00 01 EF
设置芯片为 DAC 输出，外接功放的应用	7E 1A 00 02 00 00 EF

1、芯片支持 DAC 和 PWM 两种输出，默认出厂芯片是 PWM 输出，也就是 13/14 两个脚直驱 8 欧姆 0.5 的扬声器

2、但是芯片也支持 DAC 输出，方便一些客户需要外接功放的应用场景。这个 dac 不能直驱耳机，驱动能力不够

3、DAC 输出是芯片的 9 脚，外接功放，直接串一个 105 的电容再接入功放即可

2、设置音频输出的指令，是带记忆的，也就是设置之后，下次上电生效。当然可以发送 0xC1 指令，恢复出厂设置，或者来回切换，都可以

3.4.17 组合播放指令[0x21]

组合播放[1,1][1,3][2,2][3,3]	7E 21 00 08 01 01 01 03 02 02 03 03 EF
组合播放[1,1][1,3][2,2][3,3][18,100]	7E 21 00 0A 01 01 01 03 02 02 03 03 12 64 EF
组合播放[1][3][6] -- 这个是内置 flash 指令	7E 21 00 06 00 01 00 03 00 02 EF

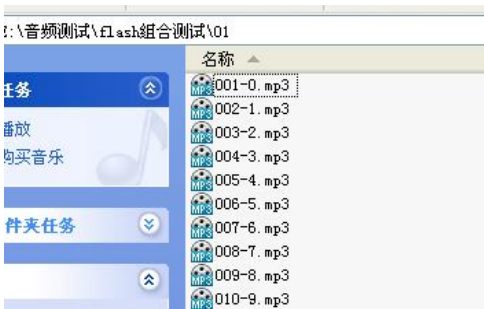

1、很多应用场合需要用到 TTS 的功能，譬如报时，报温度，报金额等等应用，如果用户拿我们的模块模仿简单的 TTS 功能的话，可能会在效果上面大打折扣，矛盾点就是在语音和语音之间的延时。直接按照一个一个文件的指定播放的话，会存在语音和语音之间的延时，这样是不能接受的。因此我们增加了组合播放的功能，同时最多支持指定播放 30 个语音，按照串口发送的顺序依次播放。

2、发送 7E 21 00 12 01 02 02 64 01 04 01 40 01 06 01 07 02 08 03 04 03 02 EF 这一帧数据，分析如下  
CMD= 0x21 --- 查阅指令表  
LEN = 0x12 = 18 个字节 ---01 02 02 64 01 04 01 40 01 06 01 07 02 08 03 04 03 02[其中一段语音，由两个参数组成，即“文件夹编号”和“文件名编号”]  
芯片会依次播放 01 文件夹第 002 曲、02 文件夹第 100 曲、01 文件夹第 004 曲、01 文件夹第 064 曲,01 文件夹第 006 曲、01 文件夹第 007 曲、02 文件夹第 008 曲、03 文件夹第 004 曲、03 文件夹第 002 曲这 9 段语音。播放完毕就停止。其中每播放完一段语音都会有串口数据返回[按照物理编号]这样就实现了跨文件夹的语音组合播放功能。  
**特别注意，这里面所有的参数是 16 进制，请用户注意，如 02 64 = 代表 02 文件夹里面的 100.mp3 这个文件**

3、在组合播放的过程中，是允许用户设置音量，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、另外在组合播放的过程中，需要停止，可以直接发送停止指令。

5、组合播放的文件，必须存放在“01”或者“02”或者其他“03-99”文件夹里面，文件必须重命名为“001xx.mp3 或者其他，如下截图。组合播放一定是要带文件夹的根目录里面是不支持的，因为根目录是不利于文件管理



6、如果组合播放发送的指定文件播放，而设备中没有对应的文件的话，组合播放会在当前停止。  
请一定让发送的指令能找到相对应的文件。一旦出错，就会停止在出错的文件位置  
在组合播放的过程中，busy 脚的电平变化可以用来检测组合播放结束

7、如果使用芯片内置的 flash 空间，则就是按照物理顺序了 7E 21 00 06 00 01 00 03 00 02 EF  
**==> 这里 00 01 00 03 00 02 会组合播放文件 1、3、2 每两个字节组成一个文件序号**

## 3.5 串口查询指令详解

一些基本的控制播放的功能，就无需查询，此章节可以跳过不看。以下我们对关键的地方进行详细的说明--查询指令：

### 3.5.1 查询当前在线的设备[0x3F]

芯片返回7E 3F 00 02 00 01 EF	代表当前的设备是 U 盘
芯片返回7E 3F 00 02 00 02 EF	代表当前的设备是 sd 卡

1、芯片在工作过程中，会不断的检测设备的在线情况，用户也可以通过 0x3F 这条指令进行查询

2、举例说明，如果芯片返回的数据为 7E 3F 02 00 0A EF

DL=0x0A = 0000 1010 代表了 TF 卡和内置 spiflash 在线

如果 DL=0x1F= 0000 1111 代表了 U 盘、TF 卡、外挂 spiflash、内置 spiflash 均在线

3、0x0F--低四位均代表一种设备。

### 3.5.2 播放状态查询指令[0x42]

正在播放	7E 42 00 02 01 01 EF	U 盘正在播放
暂停播放	7E 42 00 02 02 02 EF	TF 卡播放过程中被暂停
停止播放	7E 42 00 02 04 00 EF	在 PC 连接下载模式
	7E 42 00 02 08 01 EF	内置 spiflash 正在播放
	7E 42 00 02 10 00 EF	芯片处于连接电脑 PC 状态

1、芯片在解码过程中会有 3 种状态对用户开放。用户可以通过指令查询获取芯片的当前状态

2、播放暂停是指，正在播放一首曲目，人为的发送指令暂停播放，

播放停止是指，一首曲目播放完毕，芯片就处于播放停止的状态

3、如果返回的数据为 7E 42 00 02 02 02 EF 代表的意义详解如下：

DH = 0x02 --- 代表的是当前是 TF 卡设备，

DL = 0x02 --- 代表的是当前“TF 卡播放过程中被暂停”

4、如果返回的数据为 7E 42 00 02 02 02 EF 代表的意义详解如下：

DH 的含义--设备之间或的关系		DL 的含义	
0x01	当前设备是 U 盘	0x00	当前处于播放停止状态
0x02	当前设备是 TF 卡	0x01	当前处于正在播放状态
0x04	当前设备外挂 spiflash	0x02	当前处于正在暂停状态
0x08	当前设备是外置 flash 模式		
0x10	当前是连接 PC 状态		

### 3.5.3 指定文件夹曲目总数查询[0x4E]

查询文件夹曲目总数	7E 4E 00 02 00 01 EF	查询01文件夹的总曲目数
	7E 4E 00 02 00 0B EF	查询11文件夹的总曲目数

1、如果用户按照我们设定的规则命名文件，“01”、“02”等等，这样就可以对这些文件夹里面的曲目总数进行查询。查询的有效文件包括 MP3 其它格式的文件忽视。

### 3.5.4 当前设备的总文件夹数目查询[0x4F]

查询文件夹总数	7E 4F 00 02 00 00 EF	查询当前设备的文件夹总数
返回文件夹总数	7E 4F 00 02 00 03 EF	有3个文件夹

1、用户可以对当前的设备进行文件夹总数的查询。我们只支持“根目录”下的文件夹的数目查询。不支持文件夹里面包含文件夹。另外请用户不要建立空的文件夹，这样会造成识别错误。

2、假如设备中有 5 个有效文件夹[文件夹里面有 MP3/WAV 文件]，一个空文件夹。那么查询文件夹的总数时，会返回有 6 个文件夹。所以建议用户不要建立空的文件夹。

3、TF 卡和 U 盘、SPIFLASH 是一样的。查询的是当前的设备，如果当前处于 U 盘播放状态，则查询到的是 U 盘内部根目录的文件夹总数

### 3.5.5 查询当前芯片的软件版本[0x46]

查询芯片当前的版本	7E 46 00 02 00 00 EF
-----------	----------------------

1、芯片的应用，会存在不断的迭代的问题，所以很多时候，芯片内部的固件是需要更新的

2、用户也可以通过这条指令，来查询当前的版本，是否有变化，当用户发送 7E 46 00 02 00 00 EF 指令给芯片的时候芯片会返回最后一次修改的月份和日期来作为高低字节

举例说明：7E 46 00 02 06 30 EF 其中的 06 代表 6 月份，30 代表的 30 号。也就是此版本是 6 月 30 号修改的

## 4. 注意事项

芯片的使用, 关键的地方做如下说明:

### 4.1 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
VIL	Low-Level Input Voltage	-0.3	-	0.3*VDD	V	VDD=3.3V
VIH	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
VOL	Low-Level Output Voltage	-	-	0.33	V	VDD=3.3V
VOH	High-Level Output Voltage	2.7	-	-	V	VDD=3.3V

详细的可以参加芯片的规格书

### 4.2 应用中的注意点

1、芯片对外的接口均是 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、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
3、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。
4、芯片在睡眠状态的电流在 25uA 左右，播放中，依据音量的大小以及外接的喇叭负载，峰值电流可以达到 250mA。对于功耗有要求，请回看 3.4.5 章节的 0x0A 指令
5、用户如果直接使用我们芯片直驱扬声器，请选择合适的喇叭即可。推荐使用 8 欧姆/0.5W。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。声音还不够大，可以外挂功放芯片
6、芯片支持 MP3 音频格式。不支持其他格式，推荐使用 MP3 格式，因为这样就更节省 FLASH 的容量。
7、我们的芯片支持 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。我们的优势就是无压缩播放和高音质，所以不太建议用户对音频进行压缩。

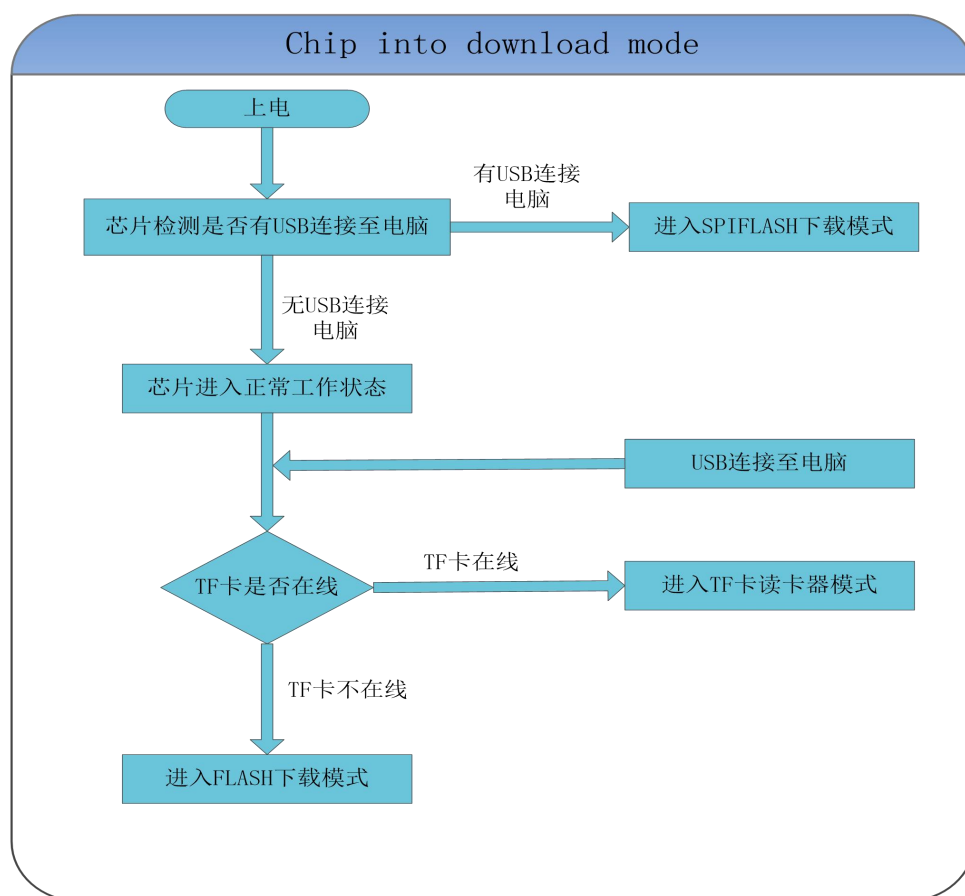
## 4.3 USB 更新语音说明

我们的芯片可以使用手机充电线直接更新语音，方便、灵活。这里分两种设备

其实 SPIFLASH 和 TF 卡，在插入 USB 连接电脑，原则上是一致的，使用的都是 MASSSTORAGE 协议。电脑会同时显示 TF 卡和 SPIFLASH 的盘符功能。这里就分为两种操作，针对芯片说明

1、芯片一上电检测到 USB 连接电脑，则进入 SPIFLASH 的读卡器功能

2、芯片上电没检测到 USB 连接电脑，进入正常工作模式。如果有 USB 连接至电脑，则进入 TF 卡读卡器模式。如果此时没有 TF 卡在线，则还是进入 SPIFLASH 的盘符模式。



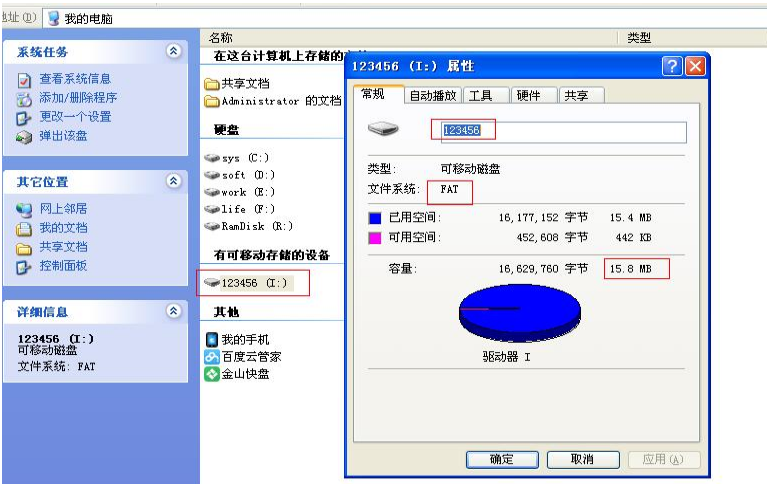


#### 4.4 USB 更新 SPIFLASH 的语音详细说明

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

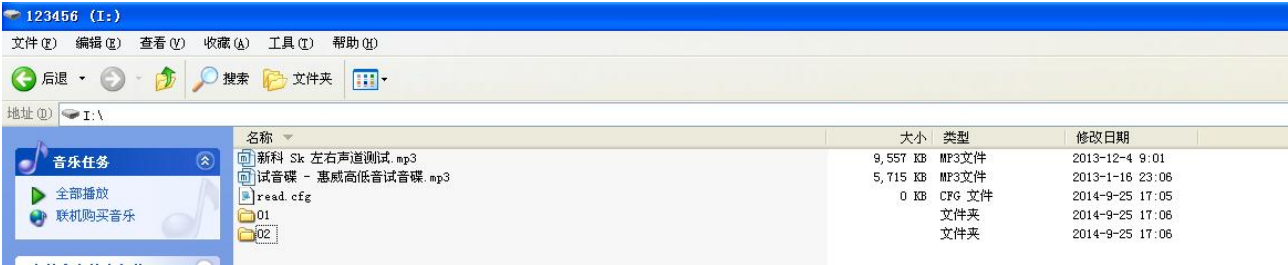
- 可以按照客户的要求，更正下载语音的窗口信息
- 无需安装任何软件，直接更新，也不需要专用下载器
- 对音质无任何压缩和损坏，保证更高的音质体验

1、插上我们模块的 USB 之后,可以以 SPIFLASH 作为存储介质的 U 盘，如下图



2、可以从上图看到 FLASH 的总容量为 15.8M 字节。已经使用的空间为 15.4M 字节。虚拟出来的设备的文件系统的为 FAT 格式。FAT 文件系统占的存储空间为 442K

3、进入设备之后，如下图



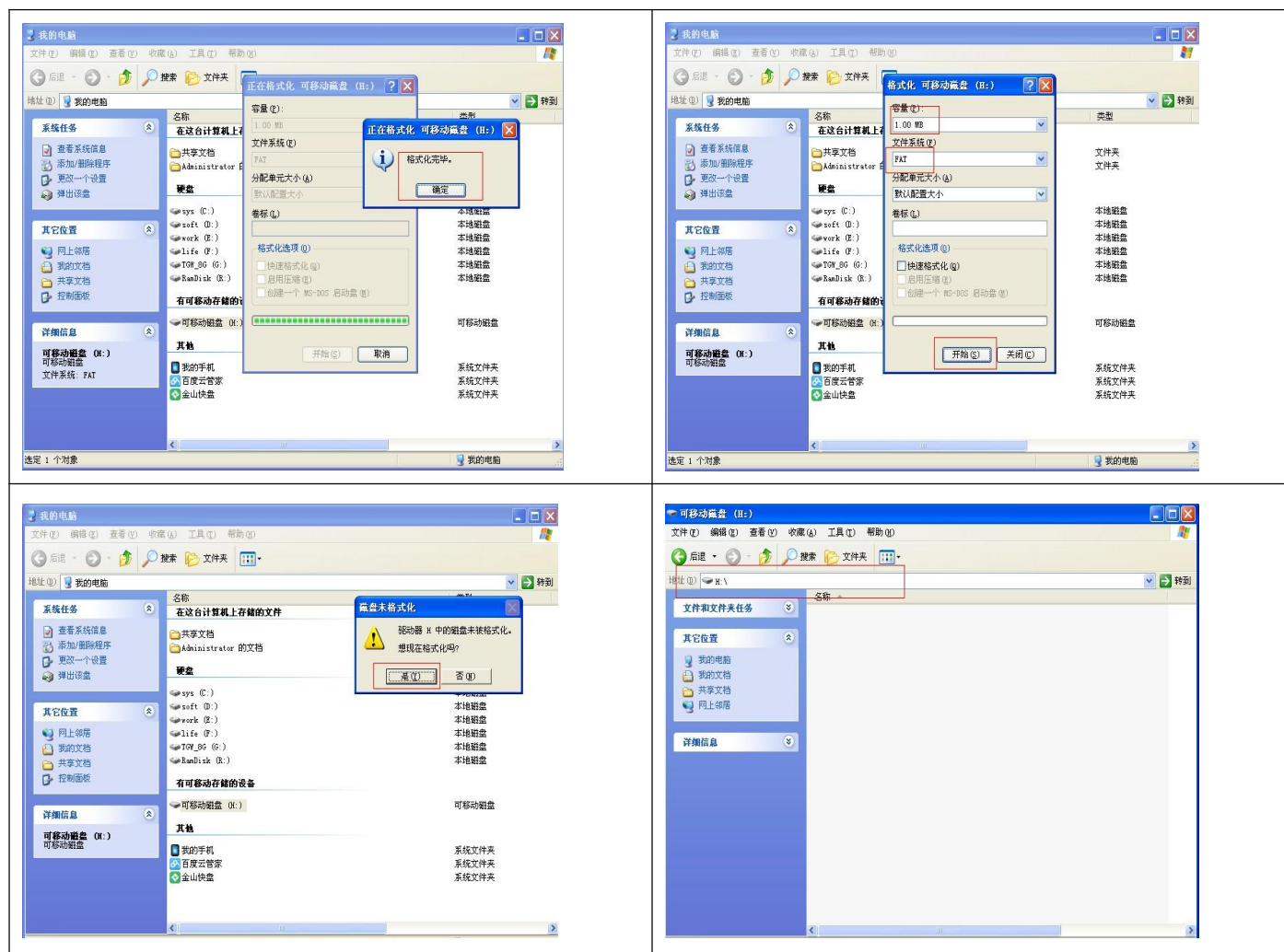
可以很清晰的看到设备里面的文件，以及文件名称。可以像操作 U 盘或者读卡器一样操作 FLASH. 只是速度会比他们慢。至于为什么后面会详细解释。

4、无论用户使用多大容量的 FLASH，我们模块都是支持的，并且内部已经做了自动识别，无需用户操心，用户也仅仅需要根据自己的需求来确定 FLASH 的容量和型号。

## 4.5 用户使用空白的 SPIFLASH 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。

- 将新的空白的 FLASH 焊接在板子上面
- 通过 USB 接口对空白的 FLASH 进行枚举和格式化
- 格式化完毕，就可以像使用 U 盘一样使用

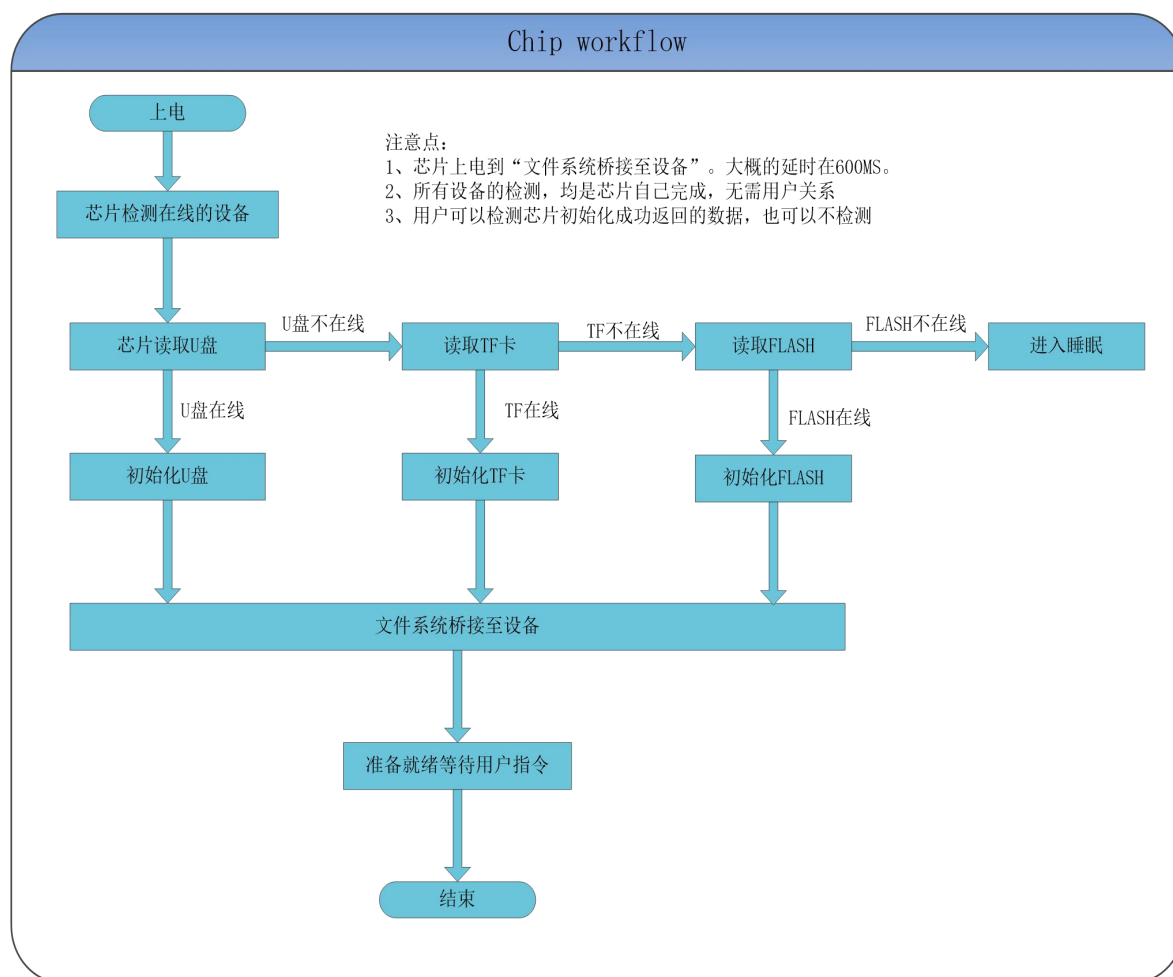


如上面的 4 个图片，就是使用空白 FLASH，FLASH 的型号为 W25Q80，容量为 1Mbyte。使用 USB 连接电脑第一次的处理过程。上面的截图可以很详细的看出步骤

## 4.6 注意事项点

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：

### 4.6.1 芯片上电的工作流程图



- 1、我司提供的所有芯片的串口部分的操作，均是一样的协议，所以不用担心不同芯片的不兼容
- 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。

---

## 4.6.2 串口编程需要适当延时的注意点

1、芯片上电之后，需要大概 1S-1.5S 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睬这些数据

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、因为芯片自带文件系统，正常情况下，在曲目不大于 1000 首的话，响应速度是低于 50ms 的  
曲目超过 3000 首之后，文件系统的切换速度会变慢一点，响应速度在 100ms --- 1S 之间不等

4、芯片内部对串口的处理是 10MS 处理一次，所以连续的指令发送时，必须要间隔 20MS 的延时。否则前面的指令将会被覆盖而得不到执行

5、如果指定文件夹文件名播放[0x0F、0x10]延时必须大于 40ms，因为芯片查找文件是需要时间的。

只要涉及到文件夹文件名查找的相关指令，40MS 的延时是必不可少的。如果芯片当前正在查找文件，串口的数据过来太频繁，会导致芯片的工作不正常

## 5. 使用芯片内置空间说明

KT142A 芯片目前分配内置的空间大小是 332KByte 。需要音质好，大概是 1 秒钟占用 2Kbyte 也就是 166 秒的样子。不限段数。音质要求不高，可以设置为 1 秒占用 1Kbyte，总时长是 332 秒 。**注意芯片内置的空间通过串口进行更换，内置空间不够，可以自己外挂 spiflash【通过 USB 连接电脑进行下载】**

芯片支持通过串口更换芯片内置空间的声音文件，需要的工具如下：

- 1、PC 端的下载工具软件---我们提供
- 2、USB 转 TTL 的串口工具---推荐使用 CH340G 的 USB 转 TTL 模块

### 5.1 MP3 文件特性介绍

总大小：320K 采样率 16KHZ 比特率 16 位 总时长：160 秒 音质：非常好 基本上是 2K 一秒	
总大小：160K 采样率 8KHZ 比特率 8 位 总时长：160 秒 音质：一般。做提示音足够 所以最多放 320 秒没问题 基本上是 1K 一秒	

所以总结下来：

- 1、需要音质好，就用 16KHZ 采样率 16 位比特率，大概需要的是 1 秒钟占用 2Kbyte 的空间。
- 2、如果音质要求不高，用 8KHZ 采样率，8 位的比特率，大概 1 秒 1Kbyte 的空间
- 3、实际芯片内置的存储空间，最多可以拿出来 332K 。意味着最好的音质，可以达到 160 秒  
注意是，不限段数，觉得不够，只能外挂一个 spiflash

### 5.2 第 1 步--产生音频文件

- 1、这个可以通过 TTS 合成软件，自己生成 mp3 或者 wav 文件 。也可以找真人录音等等方式
  - 2、获取了这些音频文件之后，自己可以试听一下效果
  - 3、音频的效果，必须是在这一步完成，包括，修音、去头去尾，调整好音量大小等等。
- 这里用到的工具，请自行网上下载，关键词：文字转语音，或者手机 app 下载 tts 文字转语音等等

5.3 第 2 步--串口工具下载音频



- 1、打开串口，默认芯片的波特率是 115200 。使用 USB 转 TTL 和电脑 PC 进行连接
- 2、芯片默认自带 1---80 数字语音，发送“播放一次”填序号，芯片就会自动播放对应序号的语音
- 3、如果需要更换芯片内置的语音，点击右上角，导入文件，再点击全部下载即可，注意文件的大小不能超过 332K，一定不能超，超了会导致芯片空间越界，导致芯片直接死机
- 4、内置空间的语音序号，是导入文件时候的序号，可以在上位机上面调整序号



---

## 5.4 内置空间的细节说明

- 1、内置的空间，是没有文件夹的说的，所以控制播放的指令，只要带有文件夹或者文件名的指令，都是不能用的
- 2、支持的指令[0x03=单文件播放一次][0x21=组合播放]等等指令
- 3、而 0x0F/0x17/0x10 这些带有文件夹属性的指令就是不支持的
- 4、内置空间的语音，如果需要批量出货的时候，默认烧写进去，可以联系我们，提交语音文件给我们即可

## 6. 免责声明

### ■ 开发预备知识

KT 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

KT 系列芯片机械结构决定了其 EMI 性能必然与一体化电路设计有所差异。KT 系列芯片的 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。

KT 系列芯片的 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 KT 系列芯片的电磁兼容特性，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

KT 工作室能保留任何时候在不事先声明的情况下对 KT 系列产品相关文档的修改权力

### ■ ESD 静电放点保护

KT 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 KT 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



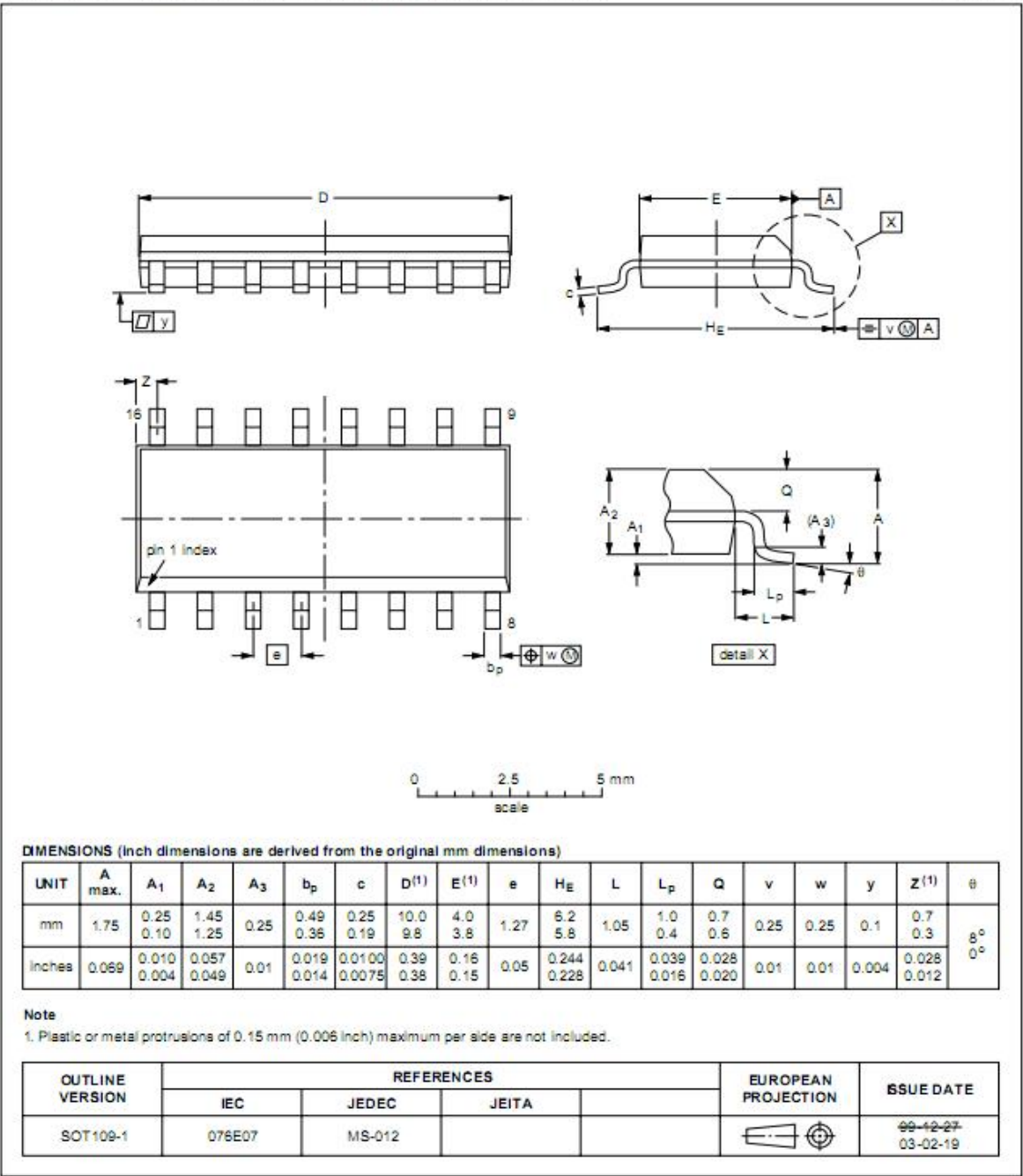




7.2 封装尺寸

封装和常用的逻辑芯片是一致的，如：74HC595、74HC138 等等 SOP16 封装的芯片。详见芯片规格书

SO16: plastic small outline package; 16 leads; body width 3.9 mm SOT109-1



## 8. 常见问题集锦

问题	1、KT142A 是什么？ 具备哪些功能？ 有什么特色？
答疑	<p>(1)、KT142A 是一颗专用于语音播报，支持外挂 SPIFLASH，或者 TF 卡、U 盘。的 MP3 语音芯片，标准 SOP16 封装，支持高音质输出。同时 KT142A 内置有 332K 的空间，可以存放 166 秒的高音质语音，不限段数，当然音质略低，可存放的长度就更长了，需要使用我们提供的上位机工具下载语音进去。</p> <p>(2)、支持串口 UART 控制，支持插播和组合播报，非常适合游戏机、播报金额</p> <p>(3)、支持虚拟 spiflash 或者 TF 卡为 U 盘，方便拷贝语音到设备</p> <p>(4)、芯片支持 DAC 输出外加功放芯片，同时内置扬声器驱动 8 欧姆/0.5W，无需外置功放</p> <p>(5)、spiflash 和 TF 卡可以同时支持，U 盘是最大支持 32G，FAT32 文件系统</p> <p>(6)、SPIFLASH 选用 W25Qxx 系列，最大支持 16M 字节</p> <p>外围电路非常简单，2 颗 105 电容即可正常工作，超宽的工作电压，支持低功耗运行</p>

问题	2、KT142A 工作电压多少？ 电流是多少？ 串口波特率是多少？ 支持多大的喇叭？
答疑	<p>(1)、芯片的工作电压是 2.6V--5V，注意如果外接功放芯片，则建议用 5V 。电压越低芯片直驱扬声器的功率就越小。</p> <p>==》如果是 5V 的系统，芯片最佳的工作电压是 4.6V 。建议 5V 串一个二极管适当的降压。二极管选择 4001 或者 4148 之类的都可以。</p> <p>==》如果是 3.3V 的系统，也没关系，只是芯片驱动扬声器会小一点声音而已，但是前提要保证电流要够，不然播放声音的时候，会把 3.3v 的供电电压拉胯导致复位</p> <p>(2)、待机电流 5mA，工作电流大概 16mA。视外挂的扬声器参数决定。需要低功耗的应用可以联系我们，因为不同的功能，对芯片功耗的处理不一样。</p> <p>==》芯片直驱 8 欧姆 0.5W 的扬声器，最大电流不会超过 200mA。以实际测试为准</p> <p>==》当然也和芯片的音量有关系。不播放的时候芯片内置的扬声器驱动会自动关闭节省功耗</p> <p>(3)、串口波特率是 115200，<b>注意是 TTL 信号。可以发指令设置波特率，详见手册描述</b></p>


	<p>(4)、使用芯片的内置功放，觉得声音还不够大，可以外置功放芯片，推荐 HAA2018 或者 8002</p> <p>详细的可以参考我们给出的原理图。最大驱动 4 欧姆 3W 的扬声器，D 类功放，详细的参见 HAA2018</p> <p>当然，HAA2018 最好是给 5V 直接供电，因为电压越高，他的带载能力越强。</p>
--	--

问题	3、KT142A 芯片如何快速的测试？如何快速的上手使用呢
答疑	<p>(1)、当您把芯片焊接到 PCB 板上面的时候，如果硬件连接正常，连接 USB 到电脑是一定能看到“虚拟的盘符”，如果没有请检查硬件的连接</p> <p>(2)、芯片的第 16 脚为测试脚，可以接一个按键，一端接地一端连接 16 脚，按一下芯片会自动播放当前设备的音乐。然后再按，就是下一段声音</p> <p>(3)、芯片的 15 脚为 busy 脚，有声音输出则为低电平，反之则为高电平</p> <p>(4)、以上都正常之后，可以使用 <b>CH340G 的 USB 转 TTL</b>，连接芯片使用“串口调试助手”进行发送指令进行测试了</p> <p>(5)、以上都正常，就可以使用单片机去正常控制了</p>

问题	4、请问资料如何下载呢？有问题需要咨询找谁呢？
答疑	<p>(1)、您好，资料在“宝贝详情”里面有专用下载链接<b>[注意要使用 PC 才能看到]</b></p> <p>(2)、无论是售前还是售后，有技术问题随时拨打“<b>13510250437 滕工</b>”</p>

问题	5、KT142A 如何下载语音到外挂的存储器里面呢？ 拷贝是按照什么顺序呢？
答疑	<p>(1)、芯片有专用的 USB 接口，可以连接 PC 电脑，芯片会自动把存储器虚拟成为 U 盘，您就可以像<b>拷贝 U 盘一样</b>，往里面拷贝语音</p> <p>(2)、<b>拷贝的顺序</b>，是先考进去的编号为 1，后考进去的编号为 2，以此类推</p> <p>(3)、还有一种是指定文件夹和文件名的方式，这样就不需要按照顺序。芯片会按照文件夹和文件名去查找对应的文件</p>

问题	6、KT142A 我焊接到 PCB 板上之后，为什么没有任何反应呢？
答疑	<p>(1)、这个问题比较模糊，所以需要多重判断，才能找出问题来</p> <p>(2)、首先看一下芯片供电是否正常，也就是芯片的 11 脚，输入电压是否是 3.3--5.2V 之间</p> <p>(3)、再看芯片的 10 脚，是否有 3.3V 的电压输出。如果超过 3.7V 以上则极有可能芯片被击穿了，只能更换了</p> <p>(4)、再连接电脑，看看电脑端是否有 USB 设备显示。同时看看 spiflash 里面是否有音乐文件</p>

问题	<p>7、KT142A 我焊接到 PCB 板上之后，为什么插入 USB 连接电脑显示如下呢？</p> 
答疑	<p>(1)、首先可以明确，就是 KT142A 和电脑之间的通讯是正常的，也就是芯片正常工作起来了</p> <p>(2)、连接电脑显示这个，不能格式化，主要的问题在于</p> <p>(3)、spiflash 选择的型号是否正确，必须选择 25Q16、等等不同容量的型号，具体看手册</p> <p>(4)、以及 spiflash 和 KT142A 的连接是否正确</p> <p>(5)、一定要注意一下：flash 要选 25Q80，25Q32，具体的可以看看我们的手册</p> <p>(6)、同时 spiflash 和 KT142A 之间的连接，有个电阻是 0 欧姆，当然短接也是可以的</p> <p>(7)、这个问题主要是硬件问题，和芯片基本没关系</p>

问题	8、KT142A 我如何使用内置的存储空间存放语音呢？
答疑	<p>1、需要借助于我们专用的 PC 端的软件工具，通过串口的形式下载进去，详见章节 5 的描述</p> <p>2、后续如果调试完毕，我们的 PC 工具端，可以打包语音文件生成固定的固件，提交给我们，进行批量的烧录即可。</p> <p>3、注意内置的空间最大只有 332Kbyte，所以下载之前请规划好内置的空间是否满足需求，如果不满足需求，则需要外挂一个合适容量的 spiflash</p> <p>4、外挂的 spiflash，是通过连接电脑，进行模拟 U 盘更换声音的</p>