



# 使用说明书

N930x MP3 解码芯片

(包括 N9300, N9301, N9302, N9303 使用手册)

N930x 芯片选型一览表

型号	封装	按键	UART 串口	一线串口	SPI FLASH	U 盘	USB 下载	TF 卡	格式	备注
N9300	SOP16	X	√	√	√	√	√	X	MP3, WAV	外挂 flash
N9301	SOP16	X	√	√	X	√	√	√	MP3, WAV	外挂 TF 卡
N9302	SOP16	ADKEY	X	X	√	√	√	X	MP3, WAV	外挂 flash
N9303	SOP16	ADKEY	X	X	X	√	√	√	MP3, WAV	外挂 TF 卡

备注: x 代表不支持。√代表支持。

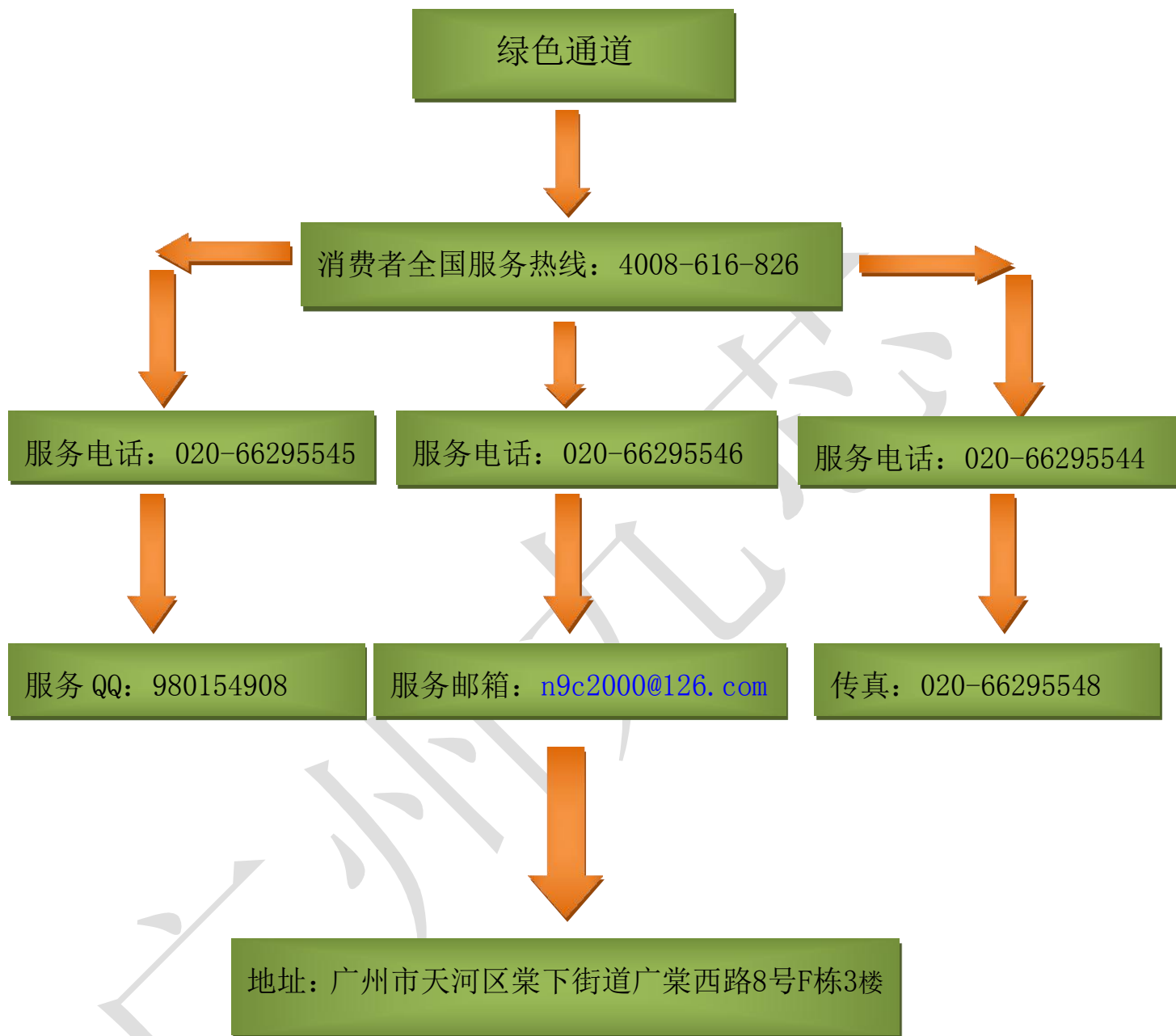
文档版本 V1.5

发布日期 2021.5.20



### 历史版本说明

版本号	日期	修改记录
V1.0	2021.04.19	初次编制
V1.4	2021.05.07	修改了电路原理图，解决断字问题，修改了芯片封装尺寸图
V1.5	2021.05.20	加入历史版本说明，完善指定路径说明



### 备注:

- 1、图片仅供参考，请以实物为准，九芯公司保留对产品外观及设计改进的权利，恕不另行通知。
- 2、广州市九芯电子为客户提供全方位的技术支持，客户可与就近的九芯电子办事或者代理机构联系，也可直接与公司总部电话联系。
- 3、本手册的最终解释权归广州市九芯电子科技有限公司所有。



目 录

历史版本说明.....	- 2 -
1.概述.....	- 5 -
1.1 简介.....	- 5 -
1.2 特征.....	- 5 -
1.3 应用场合.....	- 6 -
2.芯片使用参数.....	- 7 -
2.1 芯片结构图.....	- 7 -
2.2 电路参数.....	- 7 -
3.芯片管脚图与管脚说明.....	- 8 -
3.1 芯片管脚图.....	- 8 -
3.1.1 N9300 芯片管脚说明.....	- 8 -
3.1.2 N9301 芯片管脚说明.....	- 9 -
3.1.3 N9302 芯片管脚说明.....	- 9 -
3.1.3 N9303 芯片管脚说明.....	- 10 -
3.2 模块管脚说明.....	- 10 -
3.2.1 N9300 模块、N9301 模块管脚说明.....	- 10 -
4.文件命名以及控制方式说明.....	- 11 -
4.1 ADKEY 按键控制.....	- 11 -
4.2 MCU 一线串口控制.....	- 13 -
4.3 文件排列与命名规则.....	- 15 -
4.3.1 一线串口时序.....	- 15 -
4.3.2 程序范例.....	- 15 -
4.4 UART 串口通信协议.....	- 16 -
4.4.1. 通信数据格式: .....	- 16 -
4.4.2. 容错要求: .....	- 17 -
4.4.3. 协议约定: .....	- 17 -
4.4.4 CMD 通信指令控制列表 (16 进制) .....	- 18 -
5 芯片参考电路图.....	- 25 -
5.1 N9300 电路参考图.....	- 25 -
5.2 N9301 电路参考图.....	- 25 -
5.3 N9302 电路参考图.....	- 26 -
5.4 N9303 电路参考图.....	- 26 -
6.模块及芯片封装尺寸图.....	- 27 -
7.文件夹路径格式说明及校验码的算法.....	- 28 -
7.1 文件夹路径要求.....	- 28 -
7.2 校验和计算方法.....	- 30 -
8.注意事项.....	- 31 -
8.1 GPIO 特性.....	- 31 -
8.2 应用中的注意事项.....	- 31 -
8.3 串口操作.....	- 32 -
8.3.1 串口操作需要延时的注意事项.....	- 32 -



8.3.2 串口编程参考的说明.....	- 33 -
8.3.3 串口编程需要适当延时的注意点.....	- 33 -
9.免责声明.....	- 33 -
9.1 开发预备知识.....	- 33 -
9.2 EMI 和 EMC.....	- 33 -
9.3 修改文档的权力.....	- 33 -
9.4 ESD 静电放点保护.....	- 33 -

## 1.概述

### 1.1 简介

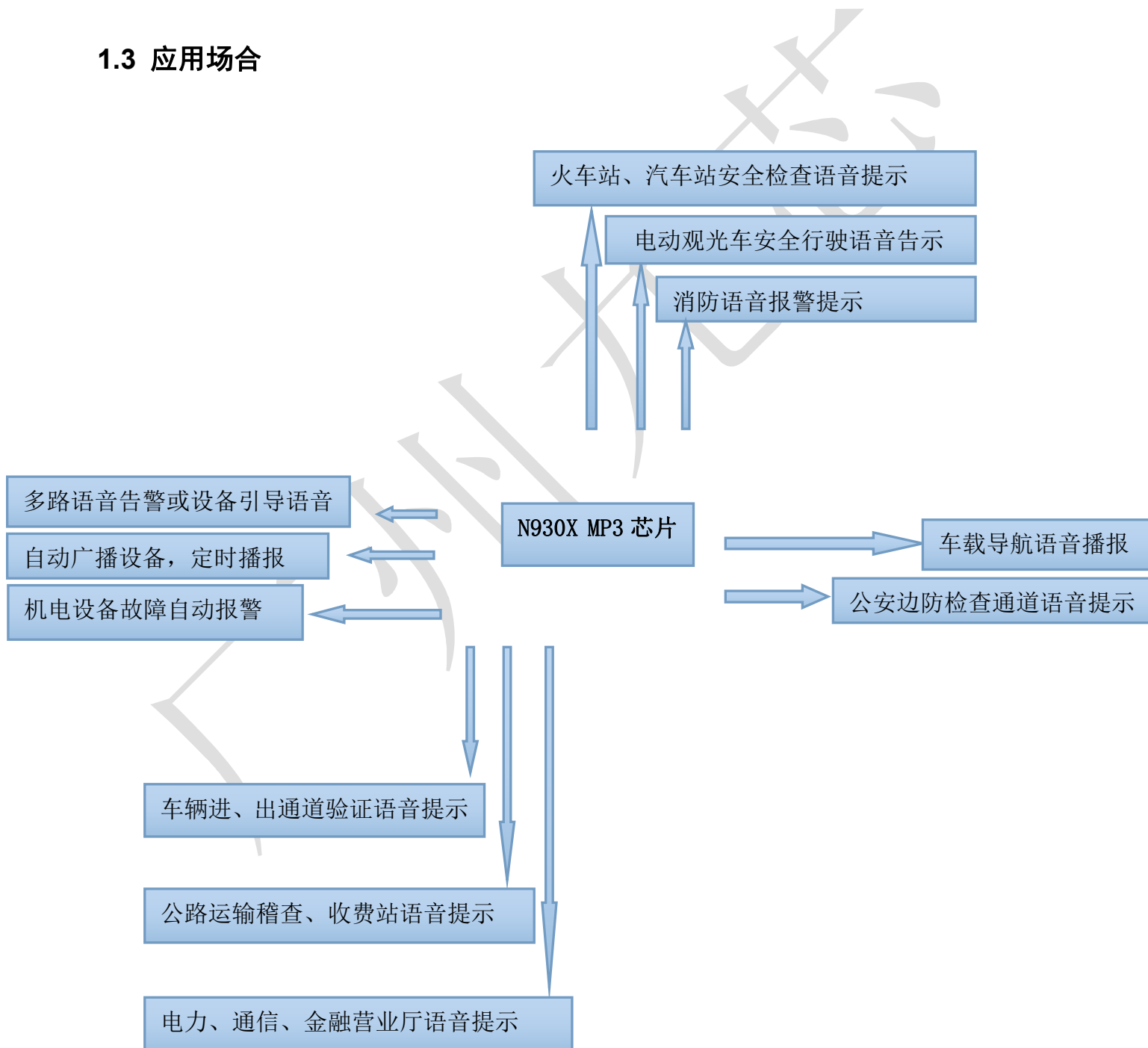
N930X 是一个提供串口的 MP3 芯片，完美的集成了 MP3、WAV 的硬解码芯片。它包括了四种功能型号的 MP3 芯片，即 N9300、N9301、N9302 和 N9303 MP3 芯片，支持 TF 卡驱动，支持电脑直接更新 spi flash 的内容，支持 FAT16、FAT32 文件系统。通过简单的 UART 串口指令或一线串口指令即可完成播放指定的音乐，以及如何播放音乐等功能，无需繁琐的底层操作，音质优美，使用方便，稳定可靠是此款产品的最大特点。另外该芯片也是深度定制的产品，专为自定义语音播放领域开发的低成本解决方案，可以根据不同的功能需要选择对应的型号，是一款精简版的也是性价比非常高的 MP3 芯片。

### 1.2 特征

- 支持采样率(KHz):8/11.025/12/16/22.05/24/32/44.1/48。音质优美，立体声。
- 24 位DAC 输出，内部采用DSP硬解码，非PWM输出，动态范围支持90dB，信噪比支持85dB
- 完全支持FAT16、FAT32 文件系统，最大支持32G的TF 卡，支持32G的U盘
- 多种控制模式，UART串口模式、一线串口模式、AD按键控制模式。
- 广播语插播功能，可以暂停正在播放的背景音乐，支持指定路径下的歌曲播放，支持跨盘符插播，支持插播提前结束
- 指定盘符播放，指定曲目播放
- 30级音量可调
- 指定路径播放（支持中英文）功能以及文件夹切换功能，指定时间段播放功能；
- 支持立体声输出播放，MP3格式，可以直推0.25W耳机喇叭；
- 支持电脑声卡控制，支持USB mass storage

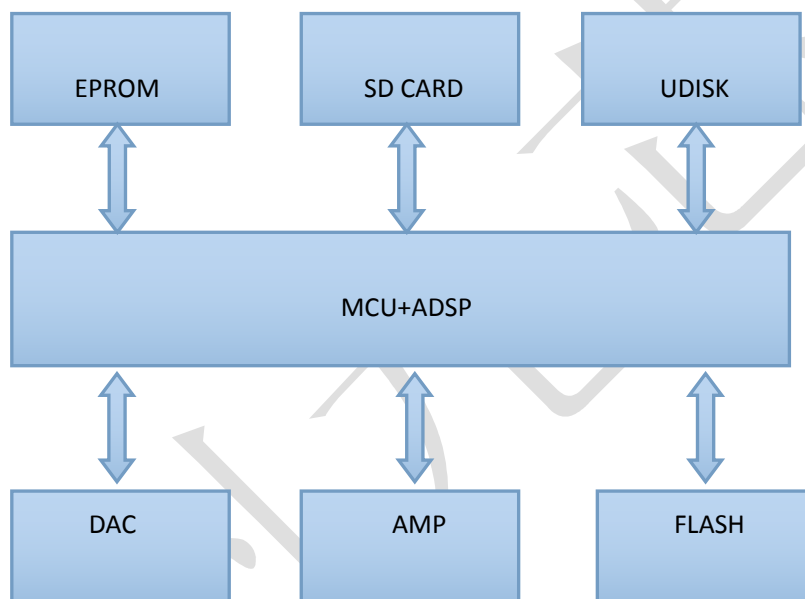
- SOP16封装形式, 外围简单;
- 宽泛的输入电源范围3.7V--5V输入, 内置看门狗复位电路, 性能稳定;
- 支持开发定制特殊功能, 请联系本公司;
- 支持红外遥控输入以及外接音频输入功能;
- 有忙信号输出功能;

### 1.3 应用场合



## 2. 芯片使用参数

### 2.1 芯片结构图



N930X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。

### 2.2 电路参数

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 (KHZ) :8/11.025/12/16/22.05/24/32/44.1/48
	3、支持Normal、Jazz、Classic、Pop、Rock 等音效
UART 接口	标准串口, TTL 电平, 波特率可设
输入电压	供电在3.7V-5V
I/O口电平	芯片I/O口电平是3.3V, 所以MCU 5V供电控制时, I/O口中间需串1K电阻
额定电流	20ma
尺寸	标准的宽体DIP16/DIP28封装



喇叭功率	8欧/3瓦
工作温度	-20度~80度
湿度	5% ~ 95%

### 3.芯片管脚图与管脚说明

#### 3.1 芯片管脚图

UART RX1	PB7	VOCM	16	VCOM
UART TX2	PB6	AGND	15	AGND
VSSIO 3	VSSIO	DACL	14	DACL
VBAT 4	VBAT DACR/PC0	DACR	13	DACR
VDDIO 5	VDDIO PA1/PC3	MCU ILINE	12	MCU ILINE
SPI DO 6	PC5 PA2/PA4	MUTE	11	MUTE
SPI CLK 7	PC4	UDP	10	USB DP
SPI CS 8	PC3/PC2	UDM	9	USB DM

N9300 MP3 芯片

UART RX1	PB7	VOCM	16	VCOM
UART TX2	PB6	AGND	15	AGND
VSSIO 3	VSSIO	DACL	14	DACL
VBAT 4	VBAT DACR/PC0	DACR	13	DACR
VDDIO 5	VDDIO PA1/PC3	MCU ILINE	12	MCU ILINE
SD CLK 6	PC5 PA2/PA4	MUTE	11	MUTE
SD CMD 7	PC4	UDP	10	USB DP
SD DAT 8	PC3/PC2	UDM	9	USB DM

N9301 MP3 芯片

PB7 1	PB7	VOCM	16	VCOM
ADKEY 2	PB6	AGND	15	AGND
VSSIO 3	VSSIO	DACL	14	DACL
VBAT 4	VBAT DACR/PC0	DACR	13	DACR
VDDIO 5	VDDIO PA1/PC3	MCU ILINE	12	MCU ILINE
SPI DO 6	PC5 PA2/PA4	MUTE	11	MUTE
SPI CLK 7	PC4	UDP	10	USB DP
SPI CS 8	PC3/PC2	UDM	9	USB DM

N9302 MP3 芯片

PB7 1	PB7	VOCM	16	VCOM
ADKEY 2	PB6	AGND	15	AGND
VSSIO 3	VSSIO	DACL	14	DACL
VBAT 4	VBAT DACR/PC0	DACR	13	DACR
VDDIO 5	VDDIO PA1/PC3	MCU ILINE	12	MCU ILINE
SD CLK 6	PC5 PA2/PA4	MUTE	11	MUTE
SD CMD 7	PC4	UDP	10	USB DP
SD DAT 8	PC3/PC2	UDM	9	USB DM

N9303 MP3 芯片

##### 3.1.1 N9300 芯片管脚说明

引脚	引脚名称	功能描述
1	UART_RX	UART串口接收
2	UART_TX	UART串口发送
3	VSSIO	电源地
4	VBAT	电源正极。3.7V~5V电压范围
5	VDDIO	3.3V输出，可给FLASH以及SD卡供电
6	SPI_DO	外挂SPI FLASH数据脚
7	SPI_CLK	外挂SPI FLASH时钟脚
8	SPI_CS	外挂SPI FLASH片选脚
9	USB_DM	USB数据口，可外接U盘以及USB读写
10	USB_DP	USB数据口，可外接U盘以及USB读写
11	BUSY	忙信号反馈输出。播放为低，待机为高
12	ONELINE	MCU一线串口输入





13	DAC_R	右声道DAC音频输出
14	DAC_L	左声道DAC音频输出
15	AGND	模拟地
16	VCOM	VCOM脚, 外接0.1uF电容到模拟地

## 3.1.2 N9301 芯片管脚说明

引脚	引脚名称	功能描述
1	UART_RX	UART串口接收
2	UART_TX	UART串口发送
3	VSSIO	电源地
4	VBAT	电源正极。3.7V~5V电压范围
5	VDDIO	3.3V输出, 可给FLASH以及SD卡供电
6	SD_CLK	外挂SD卡时钟脚
7	SD_CMD	外挂SD卡CMD脚
8	SD_DAT	外挂SD卡数据脚
9	USB_DM	USB数据口, 可外接U盘以及USB读写
10	USB_DP	USB数据口, 可外接U盘以及USB读写
11	BUSY	忙信号反馈输出。播放为低, 待机为高
12	ONELINE	MCU一线串口输入
13	DAC_R	右声道DAC音频输出
14	DAC_L	左声道DAC音频输出
15	AGND	模拟地
16	VCOM	VCOM脚, 外接0.1uF电容到模拟地

## 3.1.3 N9302 芯片管脚说明

引脚	引脚名称	功能描述
1	PB7	LINEIN音频输入脚(可功能定制IO)
2	ADKEY	AD按键输入
3	VSSIO	电源地
4	VBAT	电源正极。3.7V~5V电压范围
5	VDDIO	3.3V输出, 可给FLASH以及SD卡供电
6	SPI_DO	外挂SPI FLASH数据脚
7	SPI_CLK	外挂SPI FLASH时钟脚
8	SPI_CS	外挂SPI FLASH片选脚
9	USB_DM	USB数据口, 可外接U盘以及USB读写
10	USB_DP	USB数据口, 可外接U盘以及USB读写
11	BUSY	忙信号反馈输出。播放为低, 待机为高
12	ONELINE	MCU一线串口输入(可定制红外脚)
13	DAC_R	右声道DAC音频输出
14	DAC_L	左声道DAC音频输出
15	AGND	模拟地



16	VCOM	VCOM脚, 外接0.1uf电容到模拟地
----	------	----------------------

### 3.1.3 N9303 芯片管脚说明

引脚	引脚名称	功能描述
1	PB7	LINEIN音频输入脚 (可功能定制IO)
2	ADKEY	AD按键输入
3	VSSIO	电源地
4	VBAT	电源正极。3.7V~5V电压范围
5	VDDIO	3.3V输出, 可给FLASH以及SD卡供电
6	SD_CLK	外挂SD卡时钟脚
7	SD_CMD	外挂SD卡CMD脚
8	SD_DAT	外挂SD卡数据脚
9	USB_DM	USB数据口, 可外接U盘以及USB读写
10	USB_DP	USB数据口, 可外接U盘以及USB读写
11	BUSY	忙信号反馈输出。播放为低, 待机为高
12	ONELINE	MCU一线串口输入 (可定制红外脚)
13	DAC_R	右声道DAC音频输出
14	DAC_L	左声道DAC音频输出
15	AGND	模拟地
16	VCOM	VCOM脚, 外接0.1uf电容到模拟地

### 3.2 模块管脚说明

U1			
1	NC	NC	16
2	DACL	SPK1	15
3	DACR	SPK2	14
4	VCC	V3.3	13
5	UART_TX	USB_DP	12
6	UART_RX	USB_DM	11
7	MCU_ILINE	NC	10
8	NC	GND	9

#### 3.2.1 N9300 模块、N9301 模块管脚说明

引脚序号	引脚名称	功能描述	备注
------	------	------	----



1	NC	无连接	
2	DACL	音频输出左声道	驱动耳机、外接功放
3	DACR	音频输出右声道	驱动耳机、外接功放
4	VCC	模块电源输入	3.7V—5V正电源输入
5	UART_TX	UART串行口	UART数据输出
6	UART_RX	UART串行口	UART数据输入
7	MCU_1LINE	一线串口输入	MCU一线控制
8	NC	无连接	
9	GND	地	电源地
10	NC	无连接	
11	USB_DM	USB数据口	USB连电脑或U盘
12	USB_DP	USB数据口	USB连电脑或U盘
13	V3.3	3.3V电源输出	3.3V电源输出
14	SPK2	喇叭+	直接驱动3W/8R以下喇叭
15	SPK1	喇叭-	直接驱动3W/8R以下喇叭
16	NC	无连接	

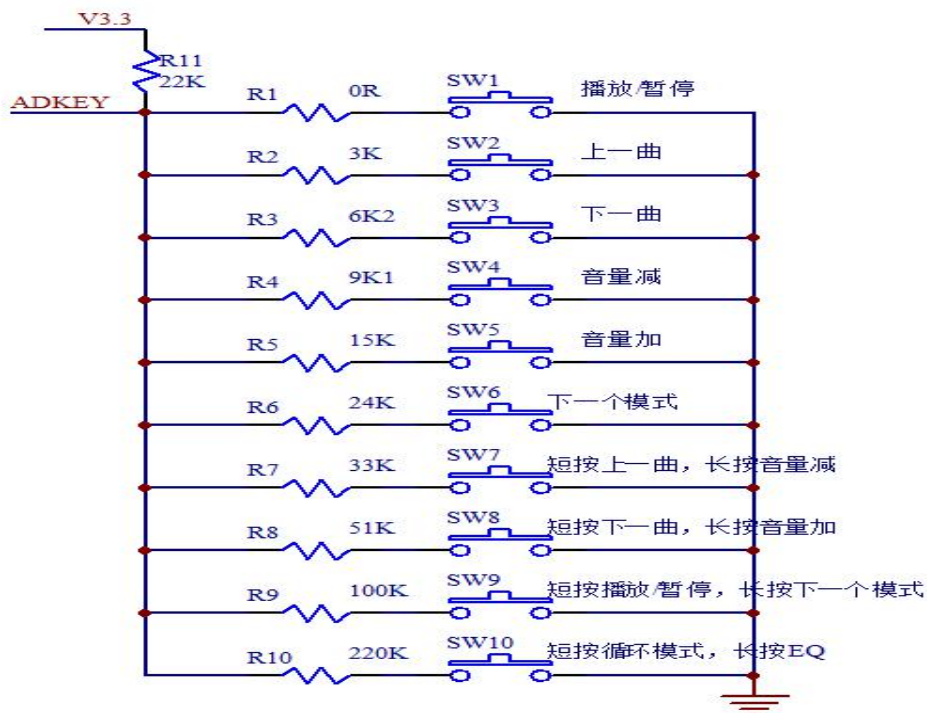
## 4.文件命名以及控制方式说明

### 4.1 ADKEY 按键控制

模块我们采用的是AD 按键的方式，取代了传统了矩阵键盘的接法，这样做的好处是充分利用了MCU 越来越强大的AD功能。设计简约而不简单，模块默认配置1个AD 口，10个按键的阻值分配，如果使用在强电磁干扰或者强感性、容性负载的场合，请参考我们的“注意事项”。

**注意：**N9300以及N9301支持一线串口和UART串口功能，不支持ADKEY功能。N9302和N9303支持ADKEY功能，不支持一线串口和UART串口。

#### (1)、ADKEY参考原理图



(2)、根据不同的电压实现不同的播放功能.

- 10个按键就取S1—S10, 电阻值就取R1—R10;
- 9个按键就取S1—S9, 电阻值就取R1—R9;
- 8个按键就取S1—S8, 电阻值就取R1—R8;
- 7个按键就取S1—S7; 电阻值就取R1—R7;
- 6个按键就取S1—S6; 电阻值就取R1—R6;
- 5个按键就取S1—S5; 电阻值就取R1—R5;
- 4个按键就取S1—S4; 电阻值就取R1—R4;
- 3个按键就取S1—S3; 电阻值就取R1—R3.

(3)、不同的阻值所对应的功能不.

- (0R) 所对应的功能有播放/暂停;
- (3K) 所对应的功能有上一曲;
- (6K2) 所对应的功能有下一曲;
- (9K1) 所对应的功能有音量减;
- (15K) 所对应的功能有音量加;
- (24K) 所对应的功能有MUSIC模式/AUX模式;
- (33K) 所对应的功能有短按上一曲, 长按音量减;
- (51K) 所对应的功能有短按下一曲, 长按音量加;
- (100K) 所对应的功能有短按播放/暂停, 长按下一个模式;



(220K) 所对应的功能有短按循环模式，长按EQ。

## 4.2 MCU—线串口控制

N930X支持单片机模拟一线串口控制，该控制方式简单方便，适用于任何单片机。

本文档中一线串口协议分为N930X和N930XA，N930X协议的一线串口加入了睡眠指令等，N930XA则和以前的N910X一线串口协议完全一样，具体指令请看指令表。

曲目指令		音量指令		播放功能	
指令	功能	指令	功能	指令	功能
0X01	选择曲目1	0XD1	睡眠模式	0XF1	切换到U盘
0X02	选择曲目2	0XD2	音量为0	0XF2	切换到SD
0X03	选择曲目3	0XD3	音量为1	0XF3	切换到flash模式
0X04	选择曲目4	0XD4	音量为2	0XF4	单曲停止
0X05	选择曲目5	0XD5	音量为3	0XF5	全部循环
0X06	选择曲目6	0XD6	音量为4	0XF6	单曲循环
0X07	选择曲目7	0XD7	音量为5	0XF7	上一曲
0X08	选择曲目8	0XD8	音量为6	0XF8	下一曲
0X09	选择曲目9	0XD9	音量为7	0XF9	上一文件夹
0X0A	选择曲目10	0XDA	音量为8	0XFA	下一文件夹
0X0B	选择曲目11	0XDB	音量为9	0XFB	播放
0X0C	选择曲目12	0XDC	音量为10	0XFC	暂停
0X0D	选择曲目13	0XDD	音量为11	0XFD	停止
0X0E	选择曲目14	0XDE	音量为12		
0X0F	选择曲目15	0XDF	音量为13		
0X10	选择曲目16	0XE0	音量为14		
0X11	选择曲目17	0XE1	音量为15		
0X12	选择曲目18	0XE2	音量为16		
0X13	选择曲目19	0XE3	音量为17		
0X14	选择曲目20	0XE4	音量为18		
0X15	选择曲目21	0XE5	音量为19		
0X16	选择曲目22	0XE6	音量为20		
0X17	选择曲目23	0XE7	音量为21		
0X18	选择曲目24	0XE8	音量为22		
0X19	选择曲目25	0XE9	音量为23		
.....	.....	0XEA	音量为24		
.....	.....	0XEB	音量为25		
.....	.....	0XEC	音量为26		
0XCC	选择曲目204	0XED	音量为27		



0XCD	选择曲目205	0XEE	音量为28		
0XCE	选择曲目206	0XEF	音量为29		
0XCF	选择曲目207	0XF0	音量为30		

备注: 最大可以控制播放到0XCF地址, 207首。

九芯MP3一线串口十六进制指令表(N930XA)

曲目指令		音量指令		播放功能	
指令	功能	指令	功能	指令	功能
0X01	选择曲目1	0XD2	音量为0	0XF1	切换到U盘
0X02	选择曲目2	0XD3	音量为1	0XF2	切换到SD
0X03	选择曲目3	0XD4	音量为2	0XF3	切换到MP3模式
0X04	选择曲目4	0XD5	音量为3	0XF4	切换到flash模式
0X05	选择曲目5	0XD6	音量为4	0XF5	下一个EQ
0X06	选择曲目6	0XD7	音量为5	0XF6	全部循环
0X07	选择曲目7	0XD8	音量为6	0XF7	单曲循环
0X08	选择曲目8	0XD9	音量为7	0XF8	上一曲
0X09	选择曲目9	0XDA	音量为8	0XF9	下一曲
0X0A	选择曲目10	0XDB	音量为9	0XFA	播放
0X0B	选择曲目11	0XDC	音量为10	0XFB	暂停
0X0C	选择曲目12	0XDD	音量为11	0XFC	停止
0X0D	选择曲目13	0XDE	音量为12	0XFD	上一文件夹
0X0E	选择曲目14	0XDF	音量为13	0XFE	下一文件夹
0X0F	选择曲目15	0XE0	音量为14		
0X10	选择曲目16	0XE1	音量为15		
0X11	选择曲目17	0XE2	音量为16		
0X12	选择曲目18	0XE3	音量为17		
0X13	选择曲目19	0XE4	音量为18		
0X14	选择曲目20	0XE5	音量为19		
0X15	选择曲目21	0XE6	音量为20		
0X16	选择曲目22	0XE7	音量为21		
0X17	选择曲目23	0XE8	音量为22		
0X18	选择曲目24	0XE9	音量为23		
0X19	选择曲目25	0XEA	音量为24		
.....	.....	0XEB	音量为25		
.....	.....	0XEC	音量为26		
.....	.....	0XED	音量为27		
0XCC	选择曲目204	0XEE	音量为28		
0XCD	选择曲目205	0XEF	音量为29		
0XCE	选择曲目206	0XF0	音量为30		
0XCF	选择曲目207				

备注: 最大可以控制播放到0XCF地址, 207首。

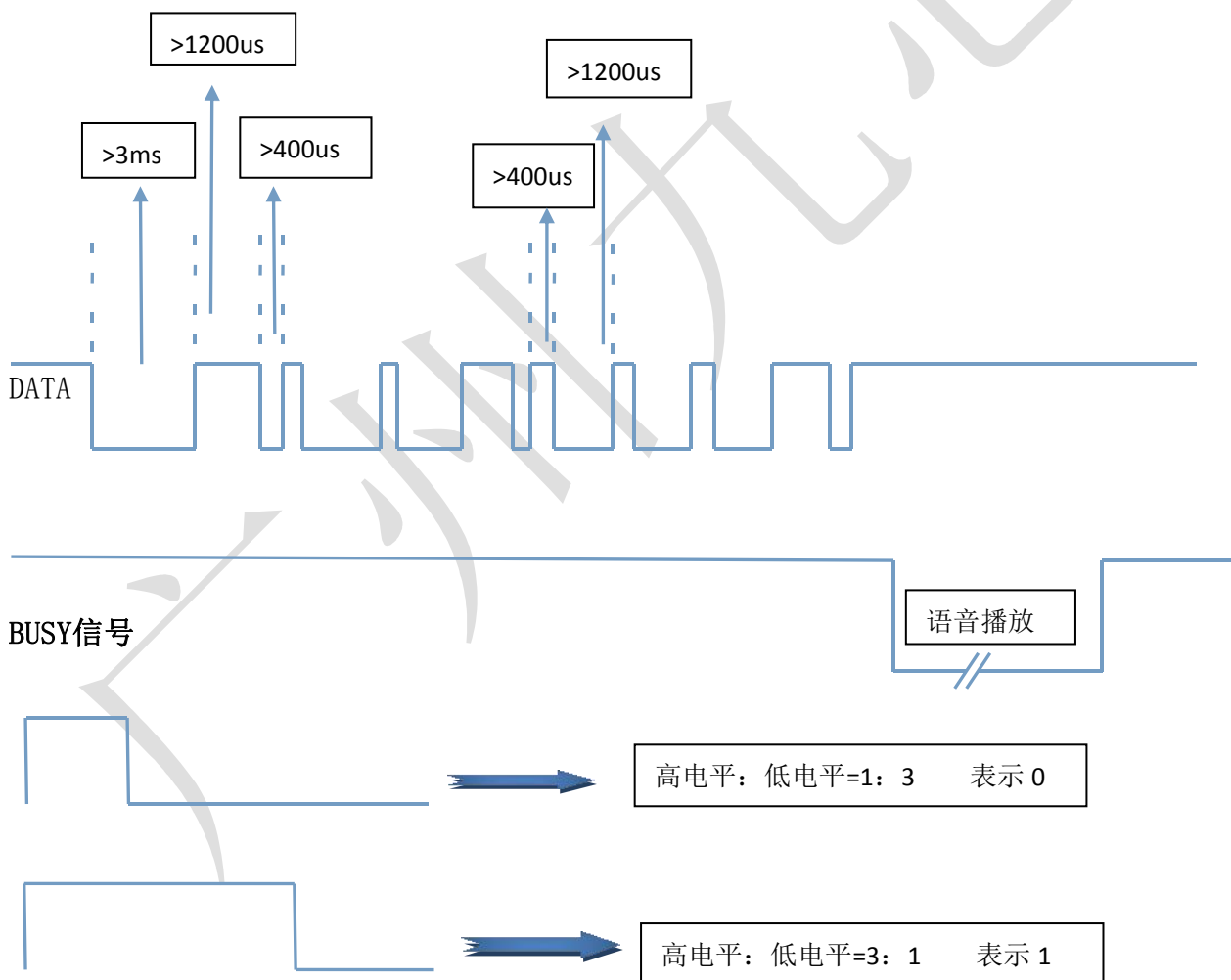


### 4.3 文件排列与命名规则

- 1.文件命名必须以 MP3 后缀结尾。
- 2.文件的顺序是按拷贝入存储设备中的先后顺序排序的。

#### 4.3.1 一线串口时序

一线串口MCU只需要一根线就可以控制了，由头码和占空比组码，**先发低位**，头码拉低3ms以上，平时保持高。总共8位。时序占空比约为**1: 3**。时序范围介于**300us: 900us**到**1000us: 3000us**之间。误差容许 10%。



#### 4.3.2 程序范例

```
(MCU: PIC16F57 晶振: 4MHz)
Sda=0; /*数据拉低*/
Wait(300); /*等待 3ms 以上*/
```

```

For(i=0;i<8;i++)          /*8 位循环*/
{
    Sda=1;                /*数据拉高*/
    If(addr&1)            /*如果地址低位拉高*/&
    {
        Wait(15);        /*等待 1200us 以上*/
        Sda=0;           /*数据拉低*/
        Wait(5);         /*等待 400us 以上*/
    }
    Else
    {
        Wait(5);         /*等待 400us 以上*/
        Sda=0;           /*数据拉低*/
        Wait(15);        /*等待 1200us 以上*/
    }
    }Addr>>=1;           /*地址右移一位*/
} Sda=1;                  /*数据拉高*/
    
```

#### 4.4 UART串口通信协议

N930X 系列内置标准 UART 异步串行接口。属于 TTL 电平接口, 可通过 MAX232 芯片转换成 RS232 电平。模块通信采用全双工串口通信, 波特率为 9600, 数据位: 8, 停止位 1 位, 检验位无。建议上电等待 1.5-2 秒后开始发送指令, 每 2 条指令间隔 280ms 以上; 设置如下图:

##### (1) 串口设置:



##### 4.4.1. 通信数据格式:

格式:【CMD】【CheckCMD】【Len】【Data1】【Data2】 DataN…… 【checksum】		
CMD	命令码	命令码, 接收的命令类型
Check CMD	命令码校验码	命令码的校验码, 即命令码的反码, 用来验证命令码





Len	数据长度	整段数据中所有数据 data 的长度, 即数据总字节个数
Data <sub>1</sub>	数据 1	该命令中所带的第 1 个字节的数据
Data <sub>2</sub>	数据 2	该命令中所带的第 2 个字节的数据
.....		.....
Data <sub>n</sub>	数据 n	该命令中所带的第 n 个字节的数据
Checksum	校验和	数据的校验和, 取 16 位校验和的低 8 位, 下面简称 SM

示例: 比如发送指定播放当前盘符下第 8 首歌曲。那就发送:

04            FB            03            06    00    08            10  
 CMD 命令    命令反码    数据长度            3 位数据            校验和

数据可以为 1 个, 也可以为多个, 主要根据 CMD 命令下规则, 对应什么数据。

#### 4.4.2. 容错要求:

接收方必须在线实时验证指令码及其反码, 及和校验的正确性。每条指令数据字节数可能不同, 因此必须实时接收到字节数后确定要接收的后续数据字节数。若有错误则应丢弃所接受指令。若一条指令接收时间超过 500ms, 则强制丢弃多接受的指令字节, 重新同步指令接收。

#### 4.4.3. 协议约定:

- 新曲目播放时会发送曲目序号。这里的序号为全盘统一编号的序号, 如果要区分父目录和子目录可以查询路径和路径深度等信息。
- 盘符拔出时会发送盘符拔出指令, 并做换盘符处理, 等待操作指令。
- 协议中所有的数据都表示 HEX(十六进制)。
- SM 校验和表示所有字节之和 (不包括校验和本身), 计算出的 16 位校验和的低 8 位。
- 协议中指的的文件都是指可以播放的文件数不包括非播放文件。
- 模式切换过去都处理停止播放状态, 要发送播放命令才会播放, 上电默认为单曲停止。
- 音量默认为最大, 可以通过相关指令来查询或设置音量大小。



#### 5.4.4 CMD 通信指令控制列表 (16 进制)

1. 播放控制: (指令: 04) 下面 SM 代表校验和

◆ 返回的播放状态代码含义: 00 已停止 01 播放中 02 已暂停

◆ 返回设备定义代码 : 00 U 盘 01 SD 02 SPI FLASH

◆ 查询播放状态 (00) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	00	00

返回: 04 FB 02 00 【播放状态代码】 SM

◆ 播放 (01) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	01	01

返回: 04 FB 02 00 【播放状态代码】 SM

◆ 暂停 (02) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	02	02

返回: 04 FB 02 00 【播放状态代码】 SM

◆ 停止 (03) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	03	03

返回: 04 FB 02 00 【播放状态代码】 SM

◆ 上一曲 (04) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	04	04

返回: (当前曲目)04 FB 03 0E 曲目高位 曲目低位 SM

◆ 下一曲 (05) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	05	05

返回: (当前曲目)04 FB 03 0E 曲目高位 曲目低位 SM

◆ 指定曲目 (06) :

指令	反码	数据长度	数据 1	数据 2	数据 3	校验和
04	FB	03	06	曲目高位	曲目低位	SM

返回: (当前曲目)04 FB 03 0E 曲目高位 曲目低位 SM



备注: 曲目一共有高低 2 个字节表示, 如果第 6 首, 就是 00 06.

$$00\ 06 = 0 * 16^3 + 0 * 16^2 + 0 * 16^1 + 6 * 16^0 = 6 \quad (\text{即第 6 首})$$

例如:

04 FB 03 06 00 08 10 指定播放当前盘符第 8 首, 曲目数从范围 1—65535, 十六进制表示为 0000H 到 FFFFH.

$$00\ 08 = 0 * 16^3 + 0 * 16^2 + 0 * 16^1 + 8 * 16^0 = 8 \quad (\text{即第 8 首})$$

◆ 当前盘符指定路径播放(07): (注意: 指定路径发送格式与以前 9100 有区别)

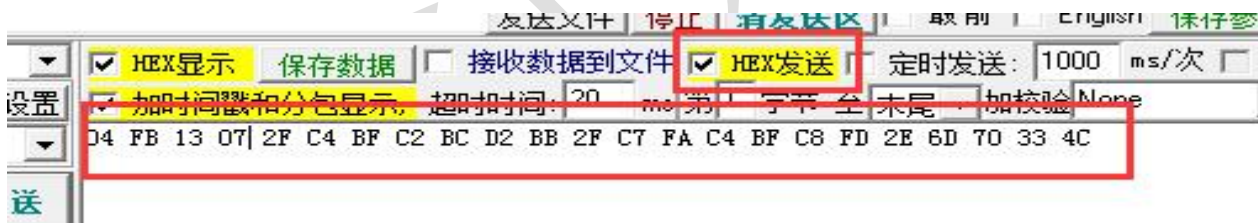
指令	反码	数据长度	数据	路径	校验和
04	FB	13 (LEN)	07	取模的数据	SM

返回: (当前曲目)04 FB 03 0E 曲目高位 曲目低位 SM

如: /目录一/曲目三.mp3

04 FB 13 07 2F C4 BF C2 BC D2 BB 2F C7 FA C4 BF C8 FD 2E 6D 70 33 4C

特别声明: 代码中每两个空一小格



数据长度

13 (LEN)这个是怎样得来的呢?

是这样子的: 13 (LEN) = 路径+1

数据

说明: 等式中的+1 就是这个 07, 意思就是说 1 个数据长度;

等式中的路径就是 2F C4 BF C2 BC D2 BB 2F C7 FA C4 BF C8 FD 2E 6D 70 33 (这些是路径)

(详见路径格式说明)

◆ 查询当前在线设备数(08) :

指令	反码	数据长度	数据 1	校验和
----	----	------	------	-----



04	FB	01	08	08
----	----	----	----	----

返回: 04 FB 02 08 在线设备数目 SM

◆ 查询当前播放设备(09) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	09	09

返回: 04 FB 02 09 盘符 SM

◆ 切换到U盘(0A) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	0A	0A

返回: 无

◆ 切换到SD(0B) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	0B	0B

返回: 无

◆ 切换到FLASH(0C) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	0C	0C

返回: 无

◆ 查询总曲目(0D) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	0D	0D

返回: 04 FB 03 0D 总曲目高位 总曲目低位 SM

◆ 查询当前曲目(0E) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	0E	0E

返回: 04 FB 03 0E 曲目高位 曲目低位 SM

◆ 上一文件夹(12) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	12	12

返回: 在播放时返回曲目序号

◆ 下一文件夹(13) : 04 FB 01 13 13

指令	反码	数据长度	数据 1	校验和
04	FB	01	13	13



返回: 在播放时返回曲目序号

◆ 结束播放(14): 04 FB 01 14 14

指令	反码	数据长度	数据 1	校验和
04	FB	01	14	14

返回: 无

◆ 短文件名获取(15):

指令	反码	数据长度	数据 1	校验和
04	FB	01	15	15

返回: 无

◆ 选曲不播放(16) : 04 FB 03 16 曲目高 曲目低 SM

返回: (当前曲目)04 FB 03 0E 曲目高位 曲目低位 SM

◆ 获取目录首曲目序号(17) (此版本暂不支持) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	17	17

返回: 04 FB 03 17 曲目高 曲目低 SM

◆ 查询当前目录下总文件数(18) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	18	18

返回: 04 FB 03 18 曲目高 曲目低 SM

## 2. 播放时间控制:(指令: 05)

◆ 总播放时间查询(00) :

指令	反码	数据长度	数据 1	校验和
05	FA	01	00	00

返回: 05 FA 04 00 时分秒 SM

◆ 当前播放时间发送(01) :

指令	反码	数据长度	数据 1	数据 2	数据 3	数据 4	校验和
05	FA	04	01	时	分	秒	SM

◆ 发送播放时间开(02) :



指令	反码	数据长度	数据 1	校验和
05	FA	01	02	02

返回: 当前时间

◆ 发送播放时间关(03) :

指令	反码	数据长度	数据 1	校验和
05	FA	01	03	03

返回: 无

### 3. 音量控制:(指令: 06)

◆ 音量最大为 30, 音量开机默认为最大, 音量范围 0 到 30 级可调。

◆ 查询音量(00) :

指令	反码	数据长度	数据 1	校验和
06	F9	01	00	00

返回: 06 F9 02 00 VOL SM

◆ 音量设置(01) :

指令	反码	数据长度	数据 1	数据 2	校验和
06	F9	02	01	VOL	SM

返回: 无

例如:

06 F9 02 01 14 16 设置音量为 20 级, 14 是 16 进制, 代表音量 20 级。

$$1*16^1+4*16^0=20(\text{级})$$

◆ 音量加(02) :

指令	反码	数据长度	数据 1	校验和
06	F9	01	02	SM

返回: 无

◆ 音量减(03) :

指令	反码	数据长度	数据 1	校验和
06	F9	01	03	03

返回: 无

### 4. 复读控制: (指令:08)

◆ 指定时间复读(00) :



指令	反码	数据长度	数据 1	数据 2	数据 3	数据 4	数据 5	校验和
08	F7	05	00	起始分	起始秒	结束分	结束秒	SM

返回: 无

例如: 08F70500020602202E 从 2 分 06 秒开始到 2 分 32 秒复读

◆ 结束复读(01) :

指令	反码	数据长度	数据 1	校验和
08	F7	01	01	01

返回: 无

执行结束命令后此次复读无效, 或执行其它操作也可以结束复读.

## 5. 插播控制: (指令: 09)

◆ 指定曲目插播(00) :

指令	反码	数据长度	数据 1	数据 2	数据 3	数据 4	校验和
09	F6	04	00	盘符	曲目高	曲目低	SM

返回: 无

例如: 09 F6 04 00 01 00 04 08 , 表示插播 SD 卡的第 4 首。

◆ 指定路径插播(01):09 F6 长度 01 盘符 路径 SM

指令	反码	数据长度	数据 1	数据 2	数据 3	校验和
09	F6	长度	01	盘符	路径	SM

说明:长度=路径长度+2

◆ 结束插播(01) : 用结束播放指令 返回: 无

## 6. 循环模式控制: (指令: 0B)

◆ 查询循环模式(00) :

指令	反码	数据长度	数据 1	校验和
0B	F4	01	00	00

返回: 0B F4 02 00 循环模式 SM

◆ 设置循环模式(01) :

指令	反码	数据长度	数据 1	数据 2	校验和
0B	F4	02	01	循环模式	SM

◆ 设置为单曲循环 : 0B F4 02 01 01 03

指令	反码	数据长度	数据 1	数据 2	校验和
0B	F4	02	01	01	03

下面列举循环播放模式代码对应的功能:



- ◆ 全部循环 : 00 按顺序播放全盘曲目, 播放完后循环播放。
- ◆ 单曲循环 : 01 一直循环播放当前曲目。
- ◆ 文件夹内循环 : 02 按顺序循环播放+当前文件夹内曲目。
- ◆ 随机播放 : 03 随机播放盘符内曲目。
- ◆ 单曲停止 : 04 播放完当前曲目一次停止。
- ◆ 顺序播放 : 05 按顺序播放全盘曲目, 播放完后停止。
- ◆ 文件夹顺序播放 : 06 顺序播放当前文件夹内曲目, 播放完停止。(此版本暂不支持)
- ◆ 文件夹随机播放 : 07 随机播放当前文件夹内曲目。(此版本暂不支持)

## 7. 睡眠指令(0C)

指令	反码	数据长度	数据 1	校验和
0C	F3	01	01	01

## 8. 返还错误信息: (指令: AA)

- ◆ AA 55 02 FF 01 01 串口接收数据错误
- ◆ AA 55 02 FF 02 02 批定的盘符找不到
- ◆ AA 55 02 FF 03 03 没有可以播放盘符。
- ◆ AA 55 02 FF 04 04 文件播放错误, 如文件找不到等

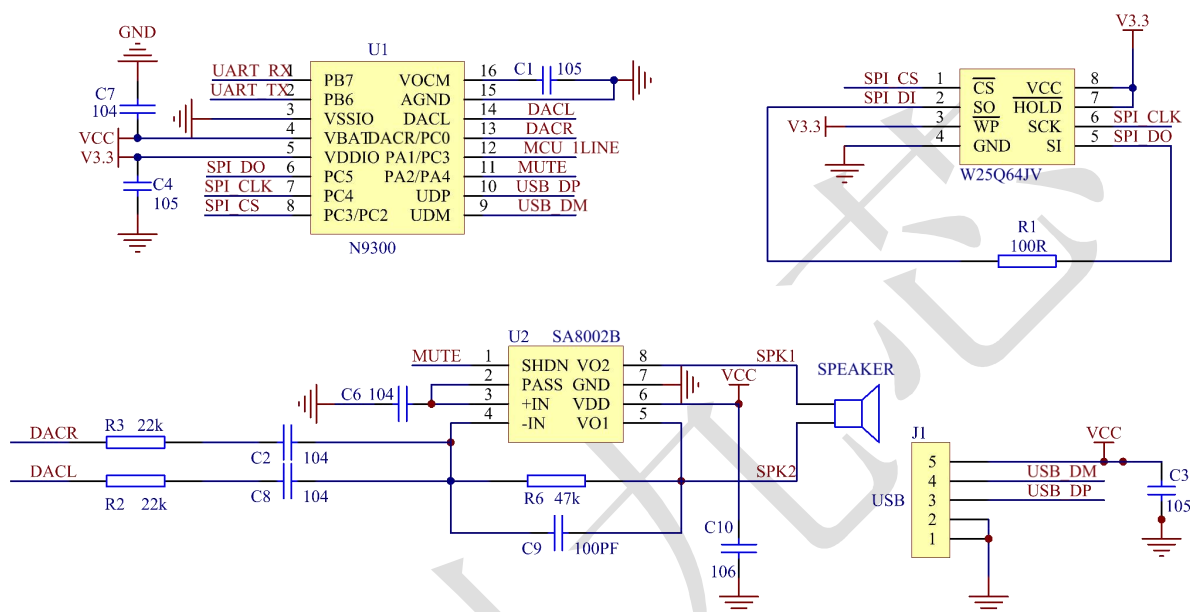
## 9. 返还设备插拔状态信息: (指令: BB)

- ◆ BB 44 01 00 00 U 盘插入
- ◆ BB 44 01 01 01 U 盘拔出
- ◆ BB 44 01 02 02 SD 卡插入
- ◆ BB 44 01 03 03 SD 卡拔出

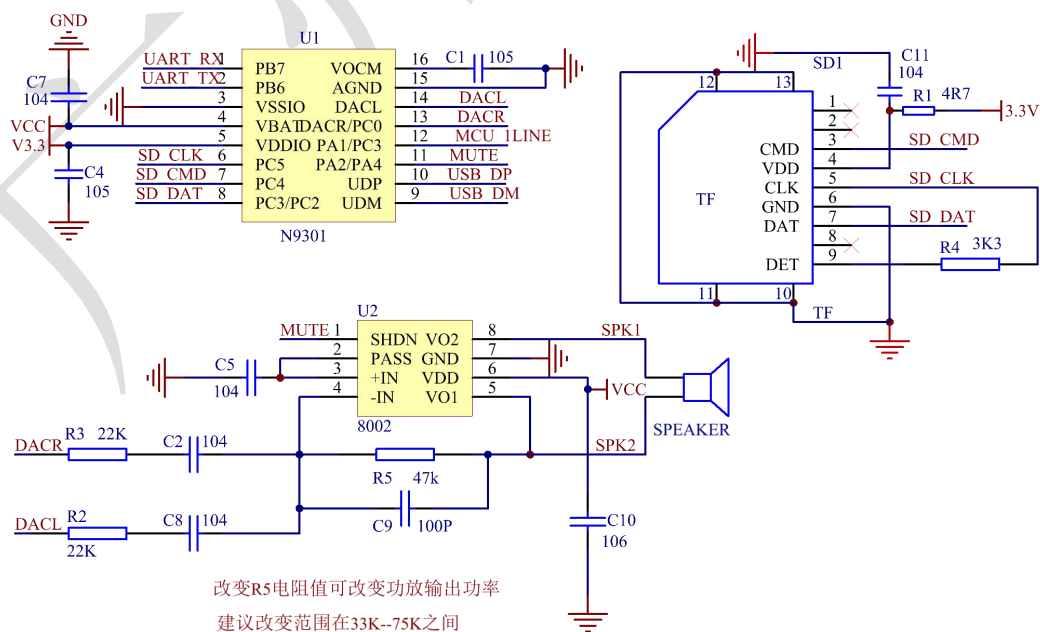


## 5. 芯片参考电路图

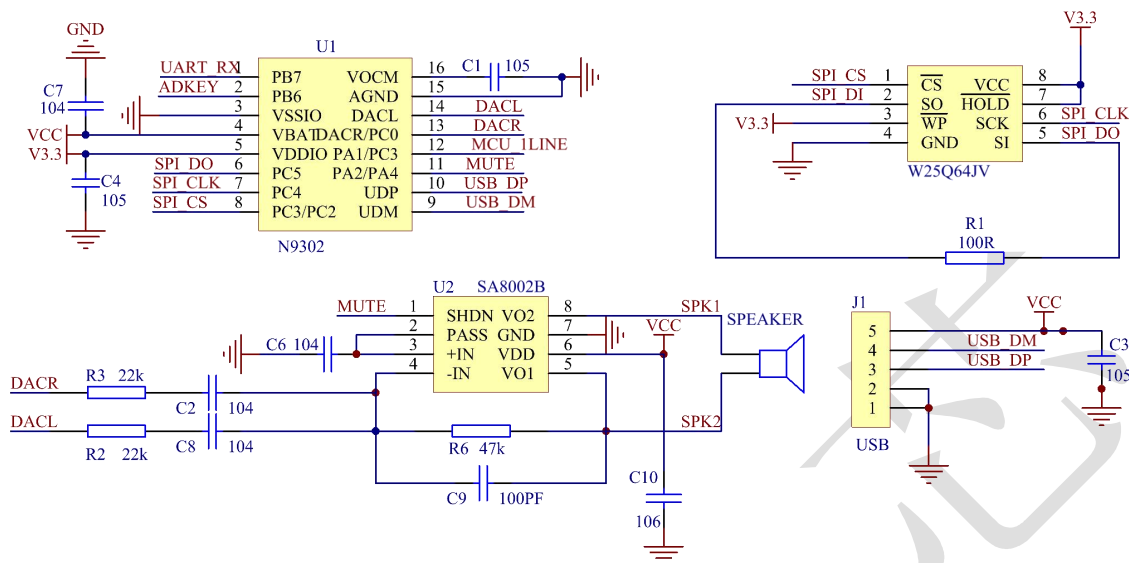
### 5.1 N9300 电路参考图



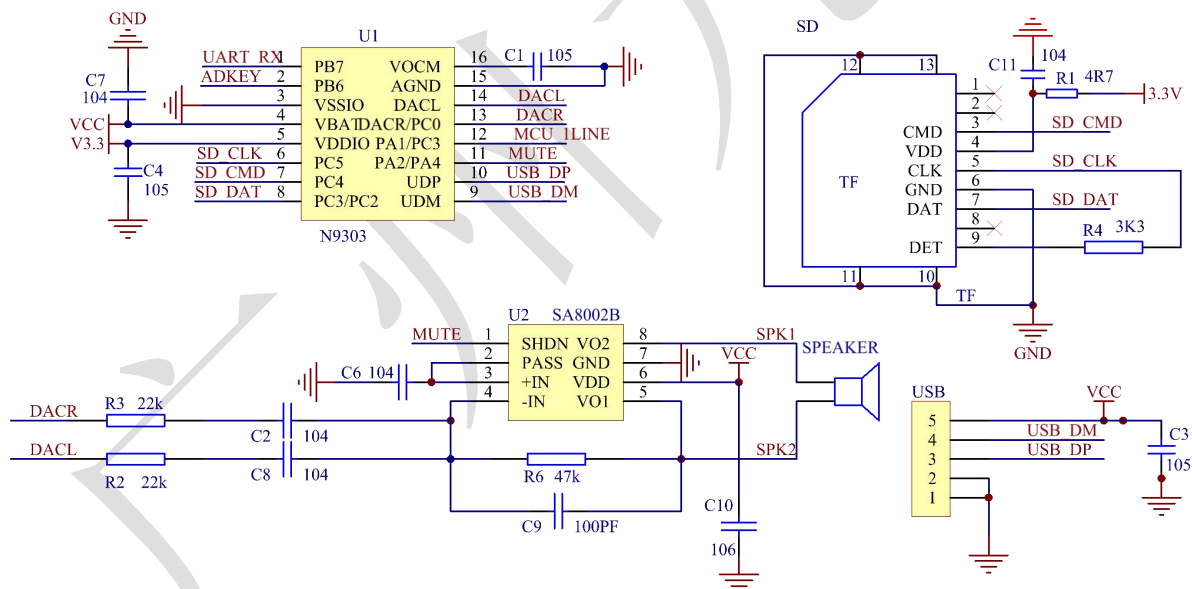
### 5.2 N9301 电路参考图



### 5.3 N9302 电路参考图



### 5.4 N9303 电路参考图

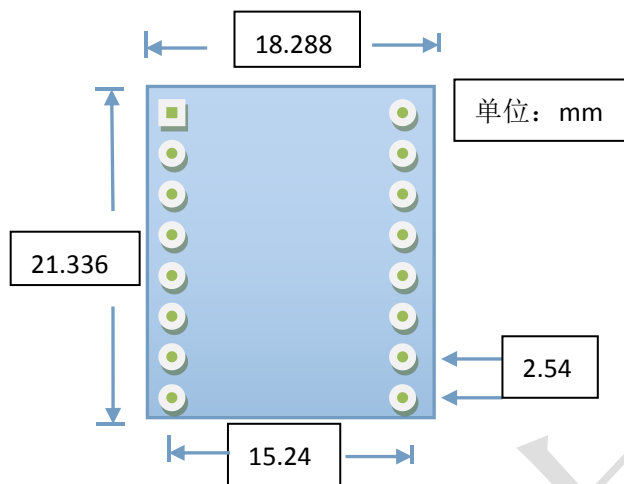


备注:

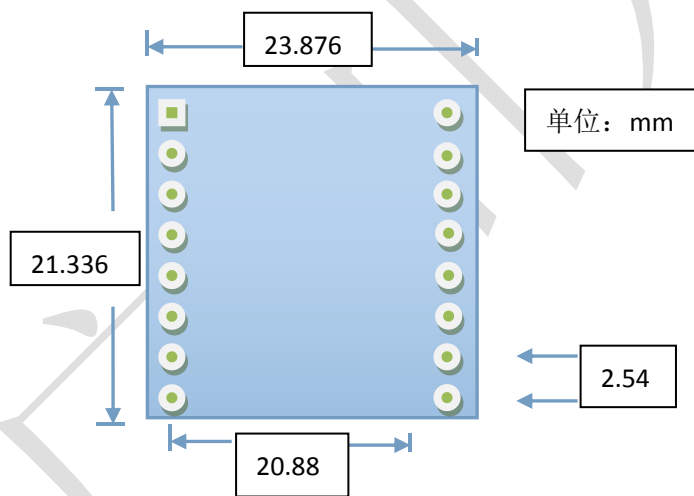
1. ADKEY 必须有 22K 上拉电阻。
2. 芯片的 IO 口不管 VCC 供电是 3.7V 还是 5V, 芯片的 IO 口始终是 3.3V, 如果 MCU 是 5V 的话, 那 MCU 与 N930X 芯片之间需要串联一个 1K 的电阻。



## 6. 模块及芯片封装尺寸图

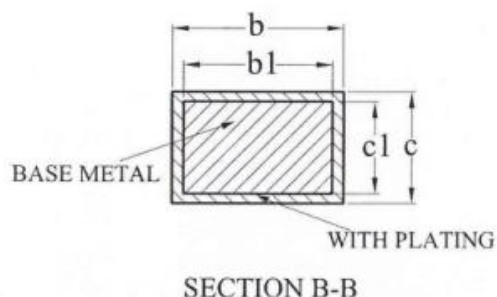
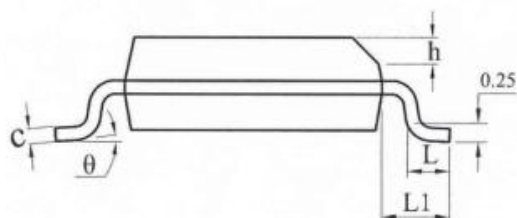
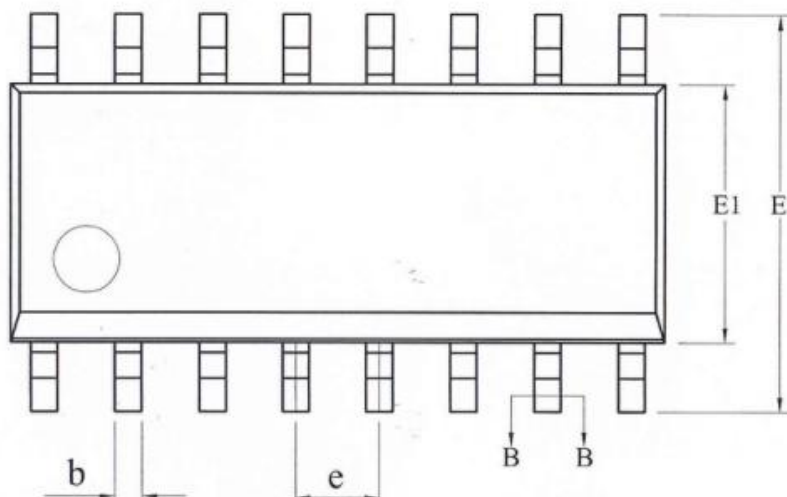
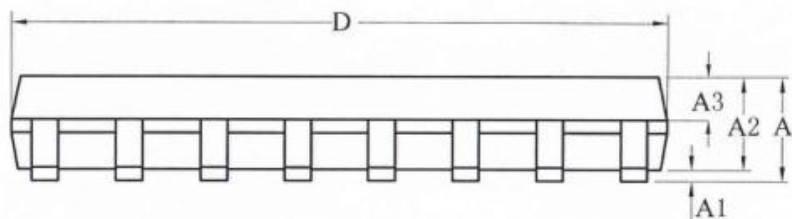


N9300 模块尺寸图



N9301 模块尺寸图

Package: SOP-16L(EP) 150mil



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	—	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	—	0.47
b1	0.38	0.41	0.44
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	9.80	9.90	10.00
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	1.27BSC		
h	0.25	—	0.50
L	0.50	—	0.80
L1	1.05REF		
theta	0	—	8°

## 7. 文件夹路径格式说明及校验码的算法

### 7.1 文件夹路径要求

- 文件夹名字最多只能是 8 个字节，8 个字节就是四个汉字或 8 个字母。
- 文件名最多也只能是 8 个字节。
- 如果文件夹名或者文件名超过 8 个字节，可以写文件名前几个字节，然后用\*号表示剩余的名字（如“/庄心妍 - 再见只是陌生人.mp3”，可以用“/庄心妍\*.mp3”表示）
- 一个汉字占用两个字节，一个空格或字符都是占用一个字节。

例如：指定文件夹下音频



/目录一/曲目三.MP3

路径数据获取:

◆ 推荐工具: amo 的编程小工具集合 V1.2.6

例如: "/目录一/曲目三.MP3", 输入格式要选择 Ascii, 在输入项里输入对应的路

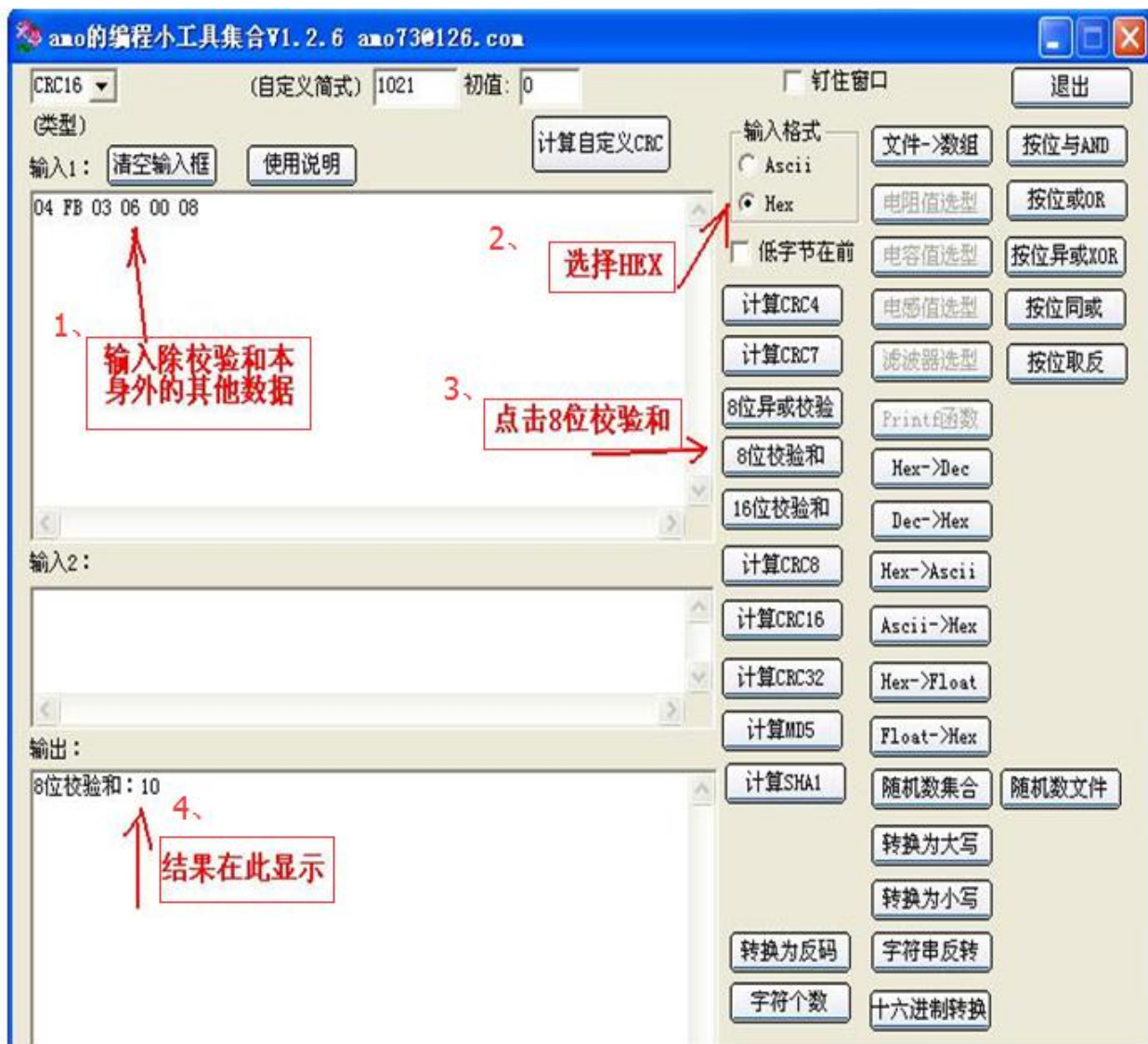
径。选择  就可以得到对应的数据。

2F C4 BF C2 BC D2 BB 2F C7 FA C4 BF C8 FD 2E 4D 50 33



## 7.2 校验和计算方法

本 UART 数据协议中校验位采样的是校验和,即将不包括校验和本身的其他字节数据累加得出的 16 位校验和,然后只取其中低 8 位。推荐使用 amo 的编程小工具集合 V1.2.6 如发送 04 FB 03 06 00 08 10 指定播放当前盘符第 8 首,那么 04 FB 03 06 00 08 将这 6 个字节的数据进行校验和,然后取低 8 位,得出校验和是如下。







## 8. 注意事项

模块的使用方法，关键的地方做如下说明：

- ◆ 模块的GPIO特性
- ◆ 应用中的注意事项
- ◆ 串口编程部分的注意

### 8.1 GPIO 特性

符号	参数	最小	典型	最大	单位	测试条件
V <sub>ol</sub>	Low-level output voltage	—	—	0.33	V	VDD=3.3
V <sub>oh</sub>	High-level output voltage	2.7	—	—	V	VDD=3.3

### 8.2 应用中的注意事项

- 1、N930X 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。
- 3、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。
- 4、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。
- 5、模块在睡眠状态的电流在 **2ua** 左右，播放 **TF** 卡，在 **15ma** 左右。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。



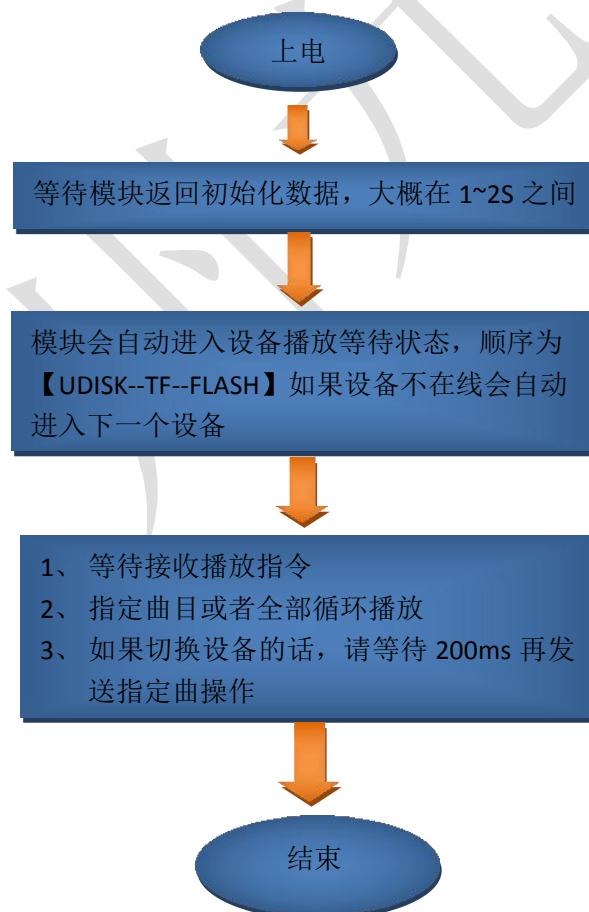
6、该模块支持 MP3、WAV、WMA 三种主流的音频格式。但是默认发货烧录的软件支持的是 MP3、WAV 这两种格式。如果特别需要支持 WMA 格式的需求，请提前说明

7、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。

### 8.3 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:
- ◆串口的操作流程
- ◆串口编程参考的说明

#### 8.3.1 串口操作需要延时的注意事项







### 8.3.2 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,第一部分是我们的测试版的测试代码,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

### 8.3.3 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 1S-1.5S 时间进行初始化的相关操作,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、因为模块自带文件系统,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

## 9.免责声明

### 9.1 开发预备知识

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。

### 9.2 EMI 和 EMC

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。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。

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。

### 9.3 修改文档的权力

本公司有能保留任何时候在不事先声明的情况下对相关文档的修改权力

### 9.4 ESD 静电放点保护

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。