



九芯电子

NINE CHIP ELECTRONICS

# N9400 系列 MP3 解码芯片 数据手册



版本号 V2.0



广州市九芯电子科技有限公司

标题

N9400 MP3 解码芯片

文档类型

数据手册

版本号

V2.0

文档摘要

本手册提供 N9400 芯片的功能特点，芯片概述以及使用简介

# 目录

1 系统概述.....	1
1.1 芯片简介.....	1
1.2 芯片选型一览表.....	1
1.3 功能特性.....	1
1.4 芯片应用场合.....	2
2 芯片管脚图与管脚说明.....	2
2.1.1 N9400 芯片管脚说明.....	2
2.1.2 N9400 芯片管脚说明.....	3
2.2 模块管脚说明.....	4
2.2.1 N9400A-16P 模块管脚图.....	3
2.2.2 N9400A-16P 模块管脚说明.....	3
2.2.3 N9400B-28P 模块管脚说明.....	3
2.2.4 N9400B-28P 模块管脚说明.....	3
3 芯片使用参数.....	5
3.1 芯片结构图.....	7
3.2 电路参数.....	7
4 文件命名以及控制方式说明.....	8
5 芯片参考电路图.....	23
6 模块及封装尺寸图.....	25
7 文件夹路径格式说明及校验码的算法.....	27

7.1 文件夹路径要求.....	27
7.2 校验和计算方法.....	28
8 注意事项.....	29
8.1 GPIO 特性.....	29
8.2 应用中的注意事项.....	29
8.3 串口操作.....	30
9 包装与运输.....	31
10 文件更新记录.....	32
11 免责声明.....	33
11.1 开发预备知识.....	33
11.2 EMI 和 EMC.....	33
11.3 修改文档的权力.....	33
11.4 ESD 静电放电保护.....	33

# 1 系统概述

## 1.1 芯片简介

N9400 是一个提供串口的 MP3 芯片,完美的集成了 MP3、WMV 的硬解码芯片。它包括了 N9400A 与 N9400B 音乐 MP3 模块,同时软件支持 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,专为固定语音播放领域开发的低成本解决方案。

## 1.2 模块选型一览表

N9400 模块选型一览表										
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
N9400A 模块	DIP16	√	√	√	√	√	√	X	MP3,WAV	
N9400B 模块	DIP24	√	√	√	√	√	√	√	MP3,WAV	

备注: x 代表不支持。√代表支持。

## 1.3 功能特性

- 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
- 有忙信号输出功能
- 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
- 指定时间播放
- 30级音量可调,5种EQ可调 (NORMAL、POP、ROCK、JAZZ、CLASSIC)
- 指定路径播放 (支持中英文)
- 支持电脑声卡控制,支持USB mass storage
- 文件夹切换
- 指定时间段复读
- 支持触发不可以打断播放

## 1.4 芯片应用场合

N9400 系列 MP3 语音解码芯片可用于各种语音提示的场合，例如：火车站、汽车站安全检查语音提示、电动观光车安全行驶语音告示、消防语音报警提示；多路语音告警或设备引导语音；多路语音告警或设备引导语音、自动广播设备，定时播报、机电设备故障自动报警；车辆进、出通道验证语音提示、公路运输稽查、收费站语音提示、电力、通信、金融营业厅语音提示；车载导航语音播报、公安边防检查通道语音提示等等。

## 2 芯片管脚图与管脚说明

### 2.1 芯片管说明

#### 2.1.1 N9400 芯片管脚图

1	PA1	PA2	24
2	IR/online	AUXL	23
3	AGND	AUXR	22
4	VCOM	USB_DP	21
5	DACL	USB_DM	20
6	DACR	ADKEY	19
7	RX	SD_DAT	18
8	TX	SDCMD	17
9	GND	SDCLK	16
10	BUSY	SPICKL	15
11	VBAT	SPI_DI/DO	14
12	VDDIO	SPI_CS	13

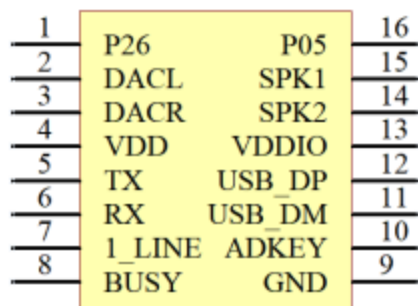
N9400 MP3主控IC

## 2.1.2 N9400 芯片管脚说明

引脚序号	引脚名称	功能描述	备注
1	PA1	PA1	可定制IO
2	IR/ONELINE	一线串口	一线串口输入 (可以定制为红外引脚)
3	AGND	模拟地	模拟地
4	VCOM	VCOM电源	VCOM电源
5	DACL	音频输出左声道	驱动耳机、外接功放
6	DACR	音频输出右声道	驱动耳机、外接功放
7	RX	UART串行口	UART数据输入
8	TX	UART串行口	UART数据输出
9	GND	地	电源地
10	BUSY	BUSY输出	播放忙信号输出
11	VBAT	电源输入	3.7V—5V正电源输入
12	VDDIO	3.3V电源输出	3.3V电源输出
13	SPI_CS	SPI_CS	接SPI FLASH CS脚
14	SPI_DI/DO	SPI_DI/DO	接SPI FLASH DO脚
15	SPI_CLK	SPI_CLK	接SPI FLASH CLK脚
16	SD_CLK	SD_CLK	接SD卡 CLK脚
17	SD_CMD	SD_CMD	接SD卡 CMD脚
18	SD_DAT	SD_DAT	接SD卡 DAT脚
19	ADKEY	ADKEY	接ADKEY按键
20	USBDM	USB数据口	USB连电脑或U盘
21	USBDP	USB数据口	USB连电脑或U盘
22	AUXR	音频输入脚	选择AUX通道可播放接入音频
23	AUXL	音频输入脚	选择AUX通道可播放接入音频
24	PA2	PA2	可定制IO

## 2.2 模块管脚说明

### 2.2.1 N9400A-16P 模块管脚图



### 2.2.2 N9400A-16P 模块管脚说明

引脚序号	引脚名称	功能描述	备注
1	P26	音频输入脚	选择AUX通道可播放接入音频
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	VDD	模块电源输入	3.7V—5V正电源输入
5	TX	UART串行口	UART数据输出
6	RX	UART串行口	UART数据输入
7	1_LINE	一线串口输入	一线串口MCU控制
8	BUSY	忙信号	有音频输出低，无音频输出高
9	GND	地	电源地
10	ADKEY	AD按键	参考AD按键电路，AD按键
11	USB_DM	USB数据口	可以接USB连电脑或U盘
12	USB_DP	USB数据口	可以接USB连电脑或U盘
13	VDDIO	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	P05	AD口	AD按键功能输入



### 2.2.3 N9400B-28P 模块管脚说明

1	NC	NC	28
2	NC	NC	27
3	NC	NC	26
4	NC	NC	25
5	NC	NC	24
6	P27	NC	23
7	P26	P05	22
8	DACL	SPK1	21
9	DACR	SPK2	20
10	VDD	VDDIO	19
11	TXD	USB_DP	18
12	RXD	USB_DM	17
13	1_LINE	ADKEY	16
14	BUSY	GND	15

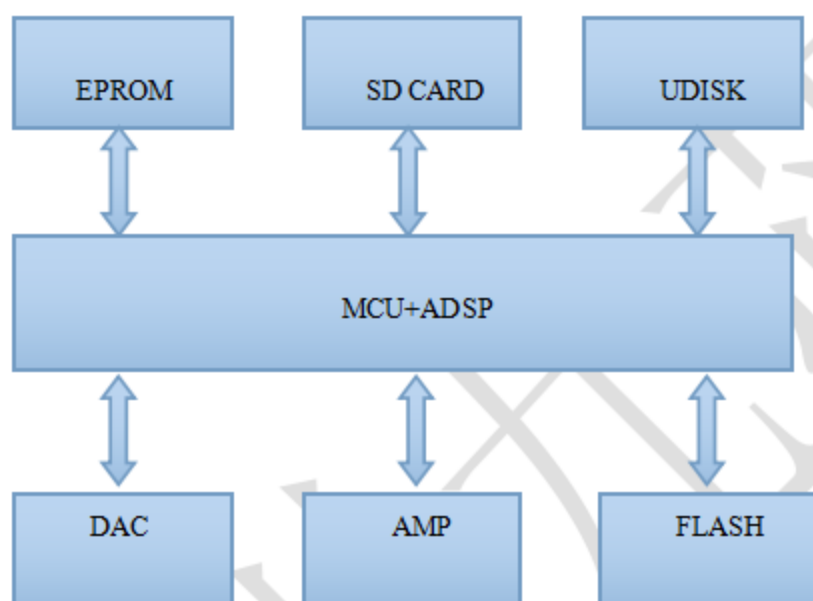
### 2.2.4 N9400B-28P 模块管脚说明

引脚序号	引脚名称	功能描述	备注
1	NC	无连接	
2	NC	无连接	
3	NC	无连接	
4	NC	无连接	
5	NC	无连接	
6	P27	音频输入脚	选择AUX通道可播放接入音频
7	P26	音频输入脚	选择AUX通道可播放接入音频
8	DACL	音频输出左声道	驱动耳机、外接功放
9	DACR	音频输出右声道	驱动耳机、外接功放
10	VDD	模块电源输入	3.7V—5V正电源输入
11	TXD	UART串行口	UART数据输出
12	RXD	UART串行口	UART数据输入
13	1_LINE	一线串口输入	MCU一线控制
14	BUSY	忙信号	有音频输出低，无音频输出高
15	GND	地	电源地
16	ADKEY	AD口	AD按键输入

17	USB_DM	USB数据口	USB连电脑或U盘
18	USB_DP	USB数据口	USB连电脑或U盘
19	VDDIO	3.3V电源输出	3.3V电源输出
20	SPK2	喇叭+	直接驱动3W/8R以下喇叭
21	SPK1	喇叭-	直接驱动3W/8R以下喇叭
22	P05	AD口	AD按键口
23	NC	无连接	
24	NC	无连接	
25	NC	无连接	
26	NC	无连接	
27	NC	无连接	
28	NC	无连接	

## 3 芯片使用参数

### 3.1 芯片结构图



N9400 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。

### 3.2 电路参数

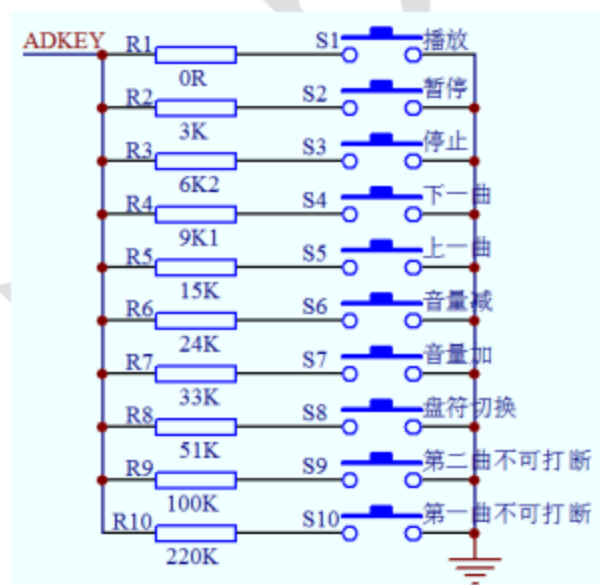
	参数
MP3文件格式	1、支持所有比特率11172-3和ISO13813-3 layer3音频解码
	2、采样率支(KHZ):8/11.025/12/16/22.05/24/32/44.1/48
	3、支持Normal、Jazz、Classic、Pop、Rock 等音效
UART 接口	标准串口, TTL 电平,波特率可设
输入电压	供电在3.7V ~ 5V
IO口电平	芯片IO口电平是3.3V, 所以MCU 5V供电控制时, IO口中间需串1K电阻
额定电流	20ma
尺寸	标准的宽体DIP16/DIP28封装
喇叭功率	8Ω/3W
工作温度	-20度 ~ 80度
湿度	5% ~ 95%

## 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2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”。

#### (1)ADKEY参考原理图



#### (2) 按键值对应表.

序号	电阻阻值	按键功能
1	0 R	播放
2	3 K	暂停
3	6K2	停止
4	9K1	上一曲
5	15 K	下一曲
6	24 K	音量减
7	33 K	音量加
8	51 K	盘符切换
9	100 K	第二曲不可打断
10	220 K	第一曲不可打断

## 4.2 MCU 一线串口控制

N9400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。

九芯MP3一线串口十六进制指令表(N9400)					
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	0XD0	进入低功耗	0XF0	音量为30
0X02	选择曲目2	0XD1	单曲播放模式	0XF1	切换到U盘
0X03	选择曲目3	0XD2	音量为0	0XF2	切换到SD
0X04	选择曲目4	0XD3	音量为1	0XF3	切换到MP3模式
0X05	选择曲目5	0XD4	音量为2	0XF4	切换到flash模式
0X06	选择曲目6	0XD5	音量为3	0XF5	下一个EQ
0X07	选择曲目7	0XD6	音量为4	0XF6	全部循环模式
0X08	选择曲目8	0XD7	音量为5	0XF7	单曲循环模式
0X09	选择曲目9	0XD8	音量为6	0XF8	上一曲
0X0A	选择曲目10	0XD9	音量为7	0XF9	下一曲
0X0B	选择曲目11	0XDA	音量为8	0XFA	播放
0X0C	选择曲目12	0XDB	音量为9	0XFB	暂停
0X0D	选择曲目13	0XDC	音量为10	0XFC	停止
0X0E	选择曲目14	0XDD	音量为11	0XFD	上一文件夹
0X0F	选择曲目15	0XDE	音量为12	0XFE	下一文件夹
0X10	选择曲目16	0XDF	音量为13		
0X11	选择曲目17	0XE0	音量为14		
0X12	选择曲目18	0XE1	音量为15		

0X13	选择曲目19	0XE2	音量为16		
0X14	选择曲目20	0XE3	音量为17		
0X15	选择曲目21	0XE4	音量为18		
0X16	选择曲目22	0XE5	音量为19		
0X17	选择曲目23	0XE6	音量为20		
0X18	选择曲目24	0XE7	音量为21		
0X19	选择曲目25	0XE8	音量为22		
.....	.....	0XE9	音量为23		
.....	.....	0XEA	音量为24		
.....	.....	0XEB	音量为25		
0XCC	选择曲目204	0XEC	音量为26		
0XCD	选择曲目205	0XED	音量为27		
0XCE	选择曲目206	0XEE	音量为28		
0XCF	选择曲目207	0XEF	音量为29		

**备注：1.最大可以控制播放到0XCF地址，207首。**

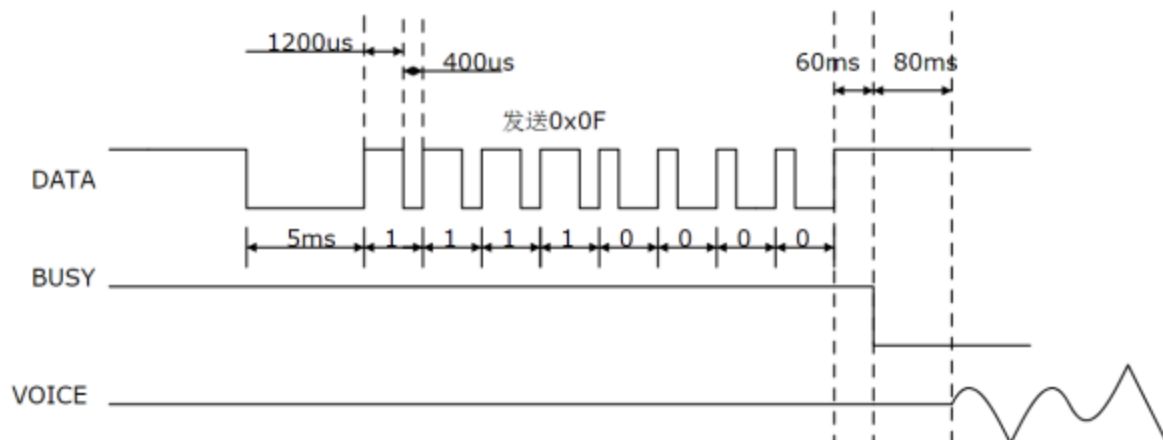
**2.进入低功耗后，需要唤醒！随意发送一个命令唤醒。**

## 一线串口时序:

一线串口MCU只需要一根信号线就可以控制了, 由头码和占空比组码, 先发低位, 头码拉低5ms, 平时保持高。

当1200us高+400us低时, 码值为1; 当400us高+1200us低时, 码值为0; 总共8位, 形成一个字节。

波形图:



DATA: 一线串口信号

BUSY (MUTE): 忙信号

VOICE: 声音输出

## 4.3 UART 串口通信协议

N9400 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 秒后开始发送指令, 每 2 条指令间隔 300ms 以上; 设置如下图:

### (1) 串口设置:



#### 4.3.1.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
CheckCMD	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
Data1	数据 1	该命令中所带的第 1 个字节的数据
Data2	数据 2	该命令中所带的第 2 个字节的数据
.....		.....
Data n	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**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**。

### 4.4.4 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 03 0E 曲目高位 曲目低位 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: 无

◆ 停止 (03) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	03	03

返回: 无

◆ 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: 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: 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: 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, (曲目不能为 0), 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):

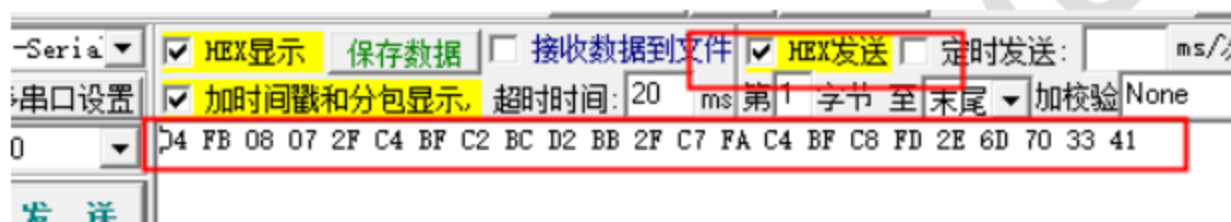
指令	反码	数据长度	数据	路径	校验和
04	FB	08 (LEN)	07	取模的数据	SM

返回: 04 FB 03 0E 曲目高位 曲目低位 SM

如: /目录一/曲目三.mp3

04 FB 08 07 2F C4 BF C2 BC D2 BB 2F C7 FA C4 BF C8 FD 2E 6D 70 33 41

特别声明: 代码中每两个空一小格



**数据长度**

08 (LEN)这个是怎样得来的呢?

是这样子的:  $08 (LEN) = \text{路径} + 1$

**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

没有找到指定的设备时返回: AA 55 02 FF 02 02

◆ 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

没有找到指定的设备时返回: AA 55 02 FF 02 02

◆ 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

◆ 上一目录(12) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	12	12

返回: 04 FB 03 0E 曲目高位 曲目低位 SM

◆ 下一目录(13) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	13	13

返回: 04 FB 03 0E 曲目高位 曲目低位 SM

◆ 结束播放 或 结束插播(14) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	14	14

返回: 04 FB 01 03 03

◆ 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: 04 FB 0C 15 短文件名(11 个字节) SM

◆ 选曲不播放(16) :

指令	反码	数据长度	数据 1	数据 2	数据 3	校验和
04	FB	03	16	曲目高	曲目低	SM

返回: 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

◆ 睡眠指令(19) :

指令	反码	数据长度	数据 1	校验和
04	FB	01	19	19

返回: 无

## 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	02

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) :

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

◆ 设置播放模式(01) :

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

返回: 无



◆ 设置全部循环模式：

指令	反码	数据长度	数据 1	数据 2	校验和
0B	F4	02	01	00	02

返回：无

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。
- ◆ 目录内循环 : 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。
- ◆ 目录顺序播放 : 06 顺序播放当前文件夹内曲目,播放完停止。
- ◆ 目录随机播放 : 07 随机播放当前文件夹内曲目。

## 7. 组合播放指令(0C)

指令	反码	数据长度	数据 1	数据 2	数据 n	校验和
0C	F3	LEN	曲目名高 8 位	曲目名低 8 位	n 曲目名低 8 位	SM

返回：无

例如：0x0C, 0xF3, 0x05, 0x01, 0x30, 0x32, 0x30, 0x33, 0xCA,表示曲目名为：02.mp3 03.mp3

组合播放。

结束组合播放：

指令	反码	数据长度	数据 1	校验和
0C	F3	01	02	02

返回：04 FB 02 00 00 01

在组合播放过程中提前结束组合播放。

## 8、AUX(P26 和 P27) 为外部声音输入通道, 增加了 DAC 输出通道

选择声道指令:

0x0A, 0xF5, 0x02, 0x01, 0x00, 0x02, 选择 DAC 输出 MP3 音乐声音;

0x0A, 0xF5, 0x02, 0x01, 0x01, 0x03, 选择 DAC 输出 AUX(P26 和 P27) 的音源;

0x0A, 0xF5, 0x02, 0x01, 0x02, 0x04, 选择 DAC 输出 MP3 和 AUX(P26、P27)混合在一起输出。

## 9、循环次数指令 (0B)

指令	反码	数据长度	数据 1	数据 2	数据 3	校验和
0B	F4	03	02	次数高	次数低	SM

返回: 无

说明: 此指令只在播放模式为全盘循环、单曲循环、目录内循环时有效, 按顺序播放。

例如: 0B F4 03 02 00 06 0A 循环 6 次; 比如当前为单曲循环, 则曲目只循环播放 6 次就会停止播放。

## 10、EQ 选择指令 (07)

指令	反码	数据长度	数据 1	数据 2	校验和
07	F8	02	01	EQ	SM

返回: 无

说明: 设置当前 EQ 只有在播放时设置有效。

EQ 值如下:

NORMAL:00; POP:01; ROCK:02; JAZZ:03; CLASSIC:04。

## 11、音乐播放结束后返回指令: 04 FB 01 03 03

说明: 组合播放是按文件名来播放组合内容的, 要求组合内部放在根目录下的 “ZH” 文件夹下, 文件名只能是两个字节, 两个数字, 或数字+字节, 或两个字母都可以, 字母不分大小写, 但在发指令时都得用大写字母。

## 12. 返回错误信息：（指令：AA）

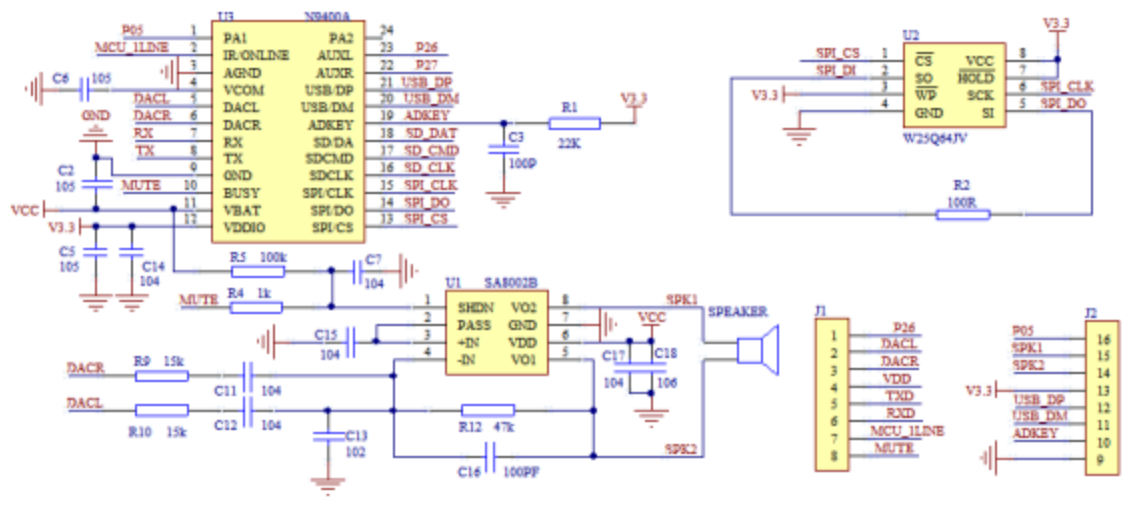
- ◆ AA 55 02 FF 01 01 串口接收数据错误
- ◆ AA 55 02 FF 02 02 指定的盘符找不到
- ◆ AA 55 02 FF 03 03 没有可以播放盘符。
- ◆ AA 55 02 FF 04 04 文件播放错误，如文件找不到等

## 13. 返回设备插拔状态信息：（指令：BB）

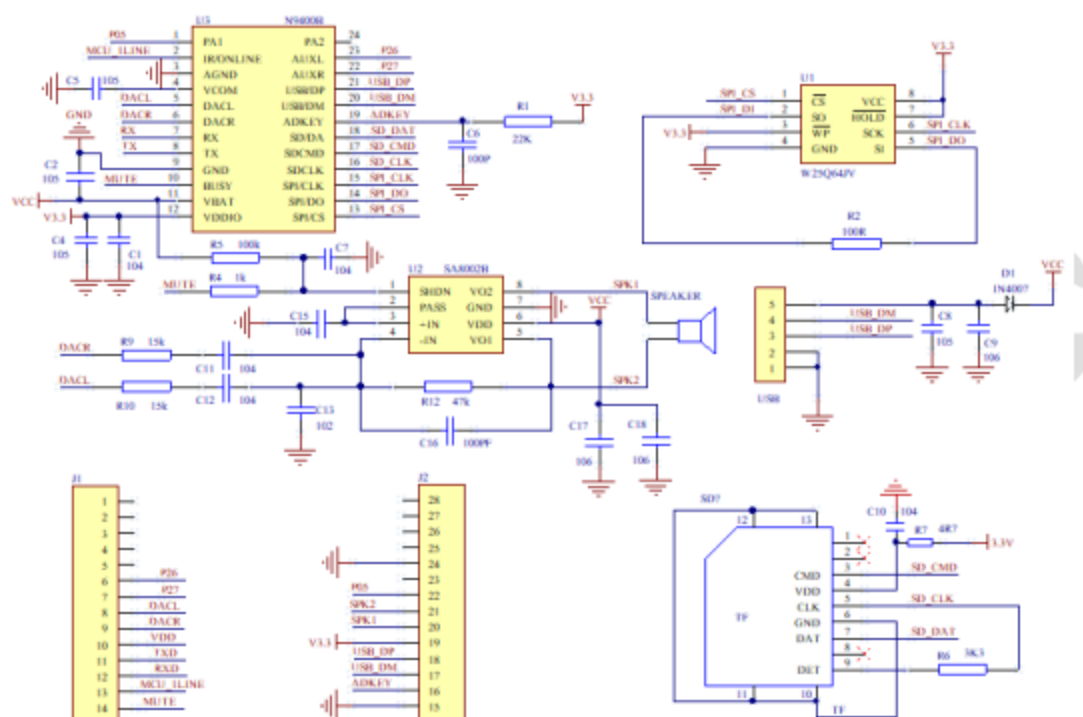
- ◆ BB 44 01 00 00 U 盘插入
- ◆ BB 44 01 01 01 U 盘拔出
- ◆ BB 44 01 02 02 SD 卡插入
- ◆ BB 44 01 03 03 SD 卡拔出

# 5 芯片参考电路图

## 5.1 N9400A 模块电路参考图



## 5.2 N9400B 模块电路参考图

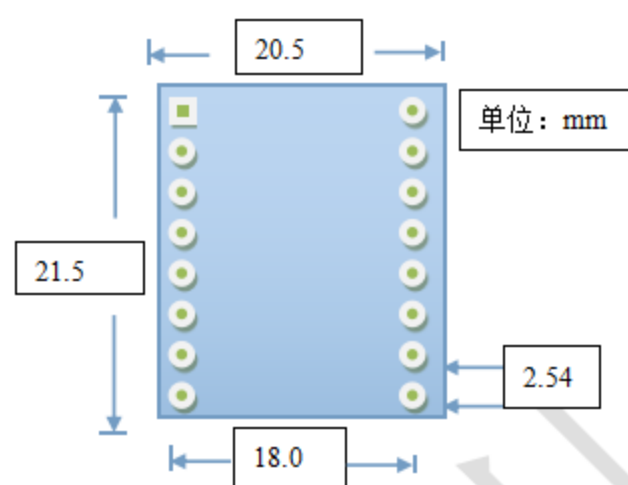


备注:

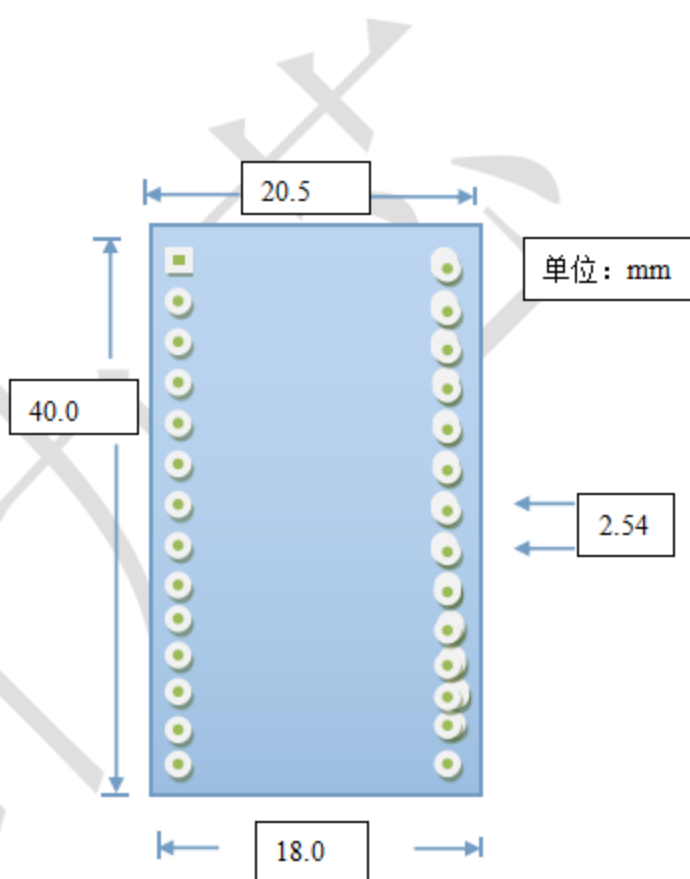
1. ADKEY 必须有 22K 上拉电阻。
2. 芯片的 IO 口不管 VCC 供电是 3.7V 还是 5V, 芯片的 IO 口始终是 3.3V, 如果 MCU 是 5V 的话, 那 MCU 与 N9400 芯片之间需要串接一个 1K 的电阻。

## 6 模块及芯片封装尺寸图

### 6.1 模块封装尺寸图

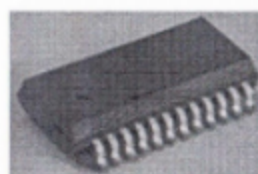
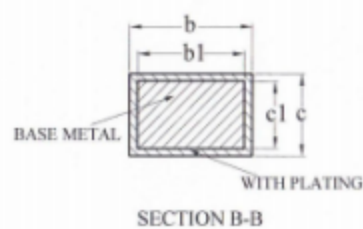
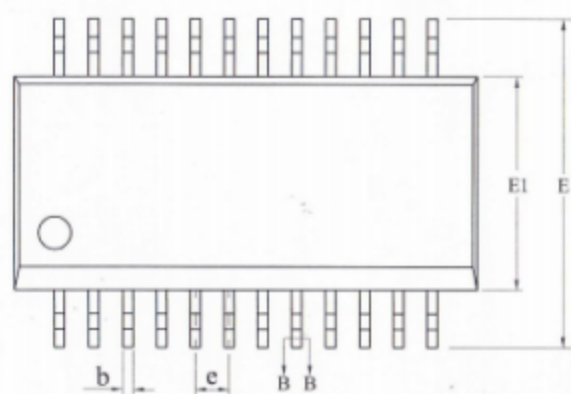
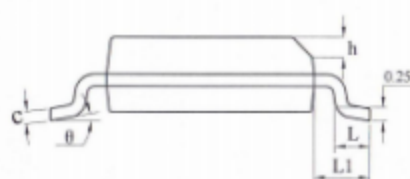
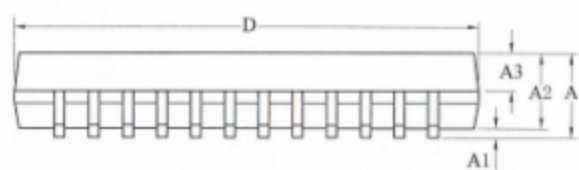


N9400A 模块尺寸图



N9400B 模块尺寸图

## 6.2 芯片封装尺寸图



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	0.15	0.25
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.23	—	0.31
b1	0.22	0.25	0.28
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	8.55	8.65	8.75
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	0.635BSC		
h	0.30	—	0.50
L	0.50	—	0.80
L1	1.05REF		
$\theta$	0	—	$g^\circ$

## 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：

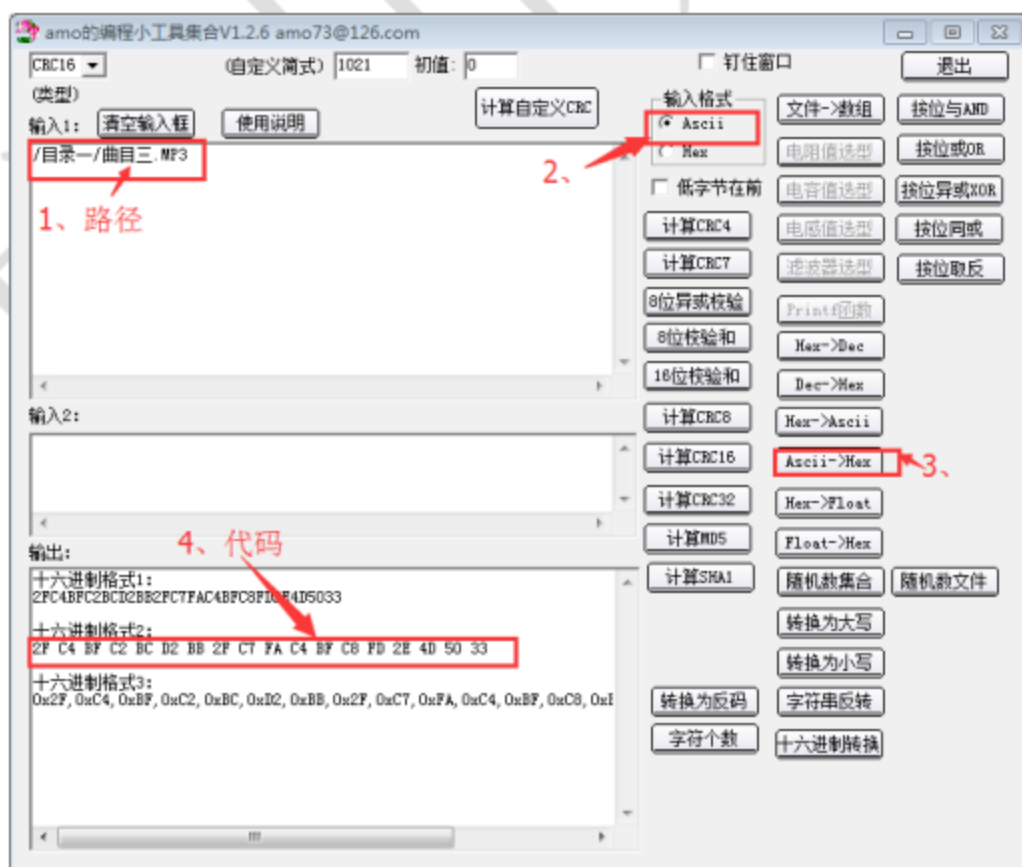
- ◆ 推荐工具：amo 的编程小工具集合 V1.2.6

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

Ascii->Hex

就可以得到对应的数据。

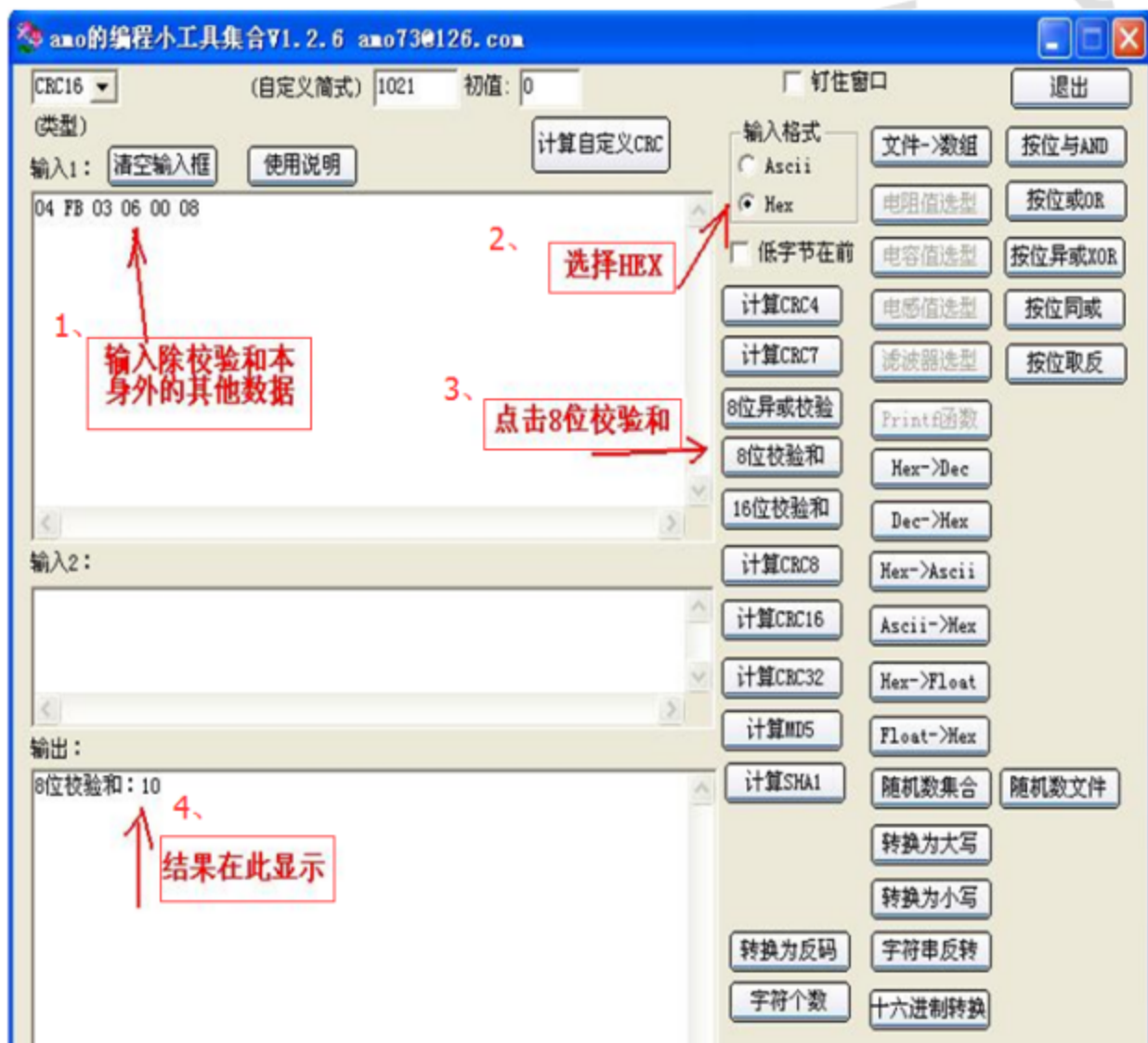
2F C4 BF C2 BC D2 BB 2F C7 FA C4 BF C8 FD 2E 4D 50 33



## 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、N9400 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 管等等。
- 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，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。

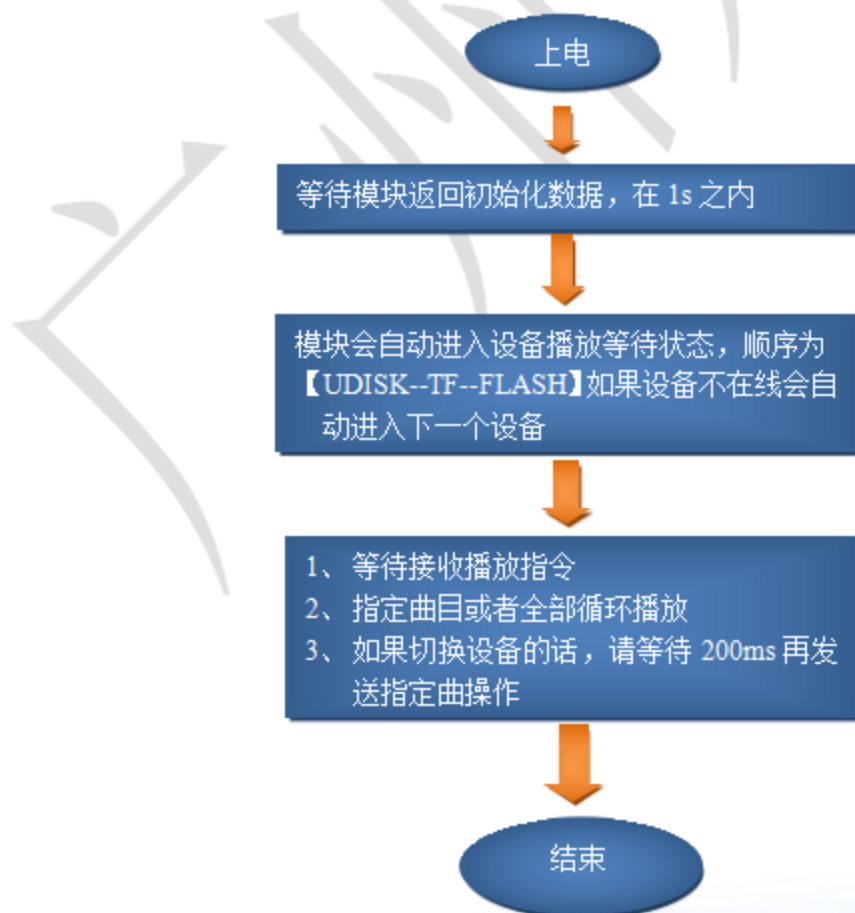
6、该模块支持 MP3、WAV、WMA 三种主流的音频格式。但是默认发货烧录的软件支持的是 MP3、WAV 这两种格式。如果特别需要支持 WMA 格式的需求，请提前说明

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

#### 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 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。

## 9 包装与运输

---

### 9.1 包装

N9400 系列芯片采用防静电防潮真空包装。

### 9.2 ESD 防护

请注意在芯片运输和生产过程中防静电和防潮



**CAUTION!** ESD SENSITIVE DEVICE!

请注意使用、包装和运输过程中的静电防护!

## 10 文件更新记录

---

版本	修订说明	日期
V2.0	重新整理串口指令	2022.02.07

# 11 免责声明

---

## 11.1 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。

## 11.2 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。

## 11.3 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

## 11.4 ESD 静电放电保护

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。