



VS1003B MP3 Board 使用手册

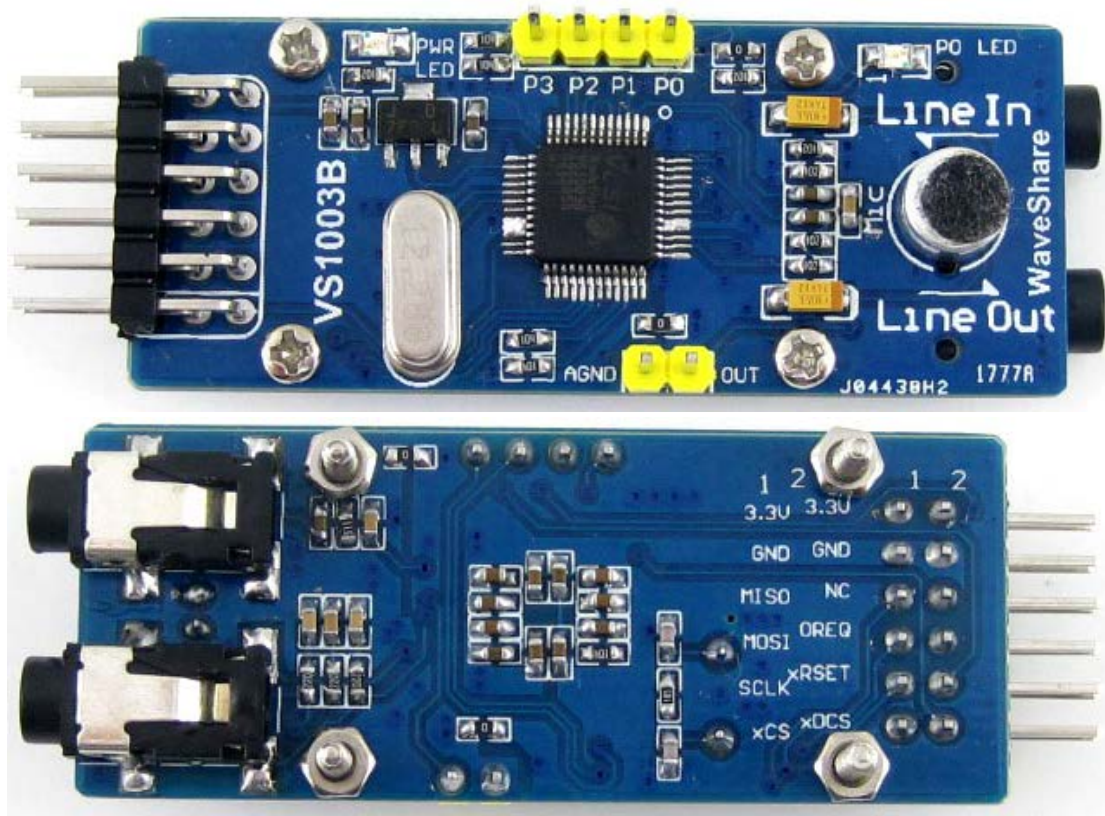
1. 特性参数	2
2. 使用说明	2
2.1 模块引脚说明	2
2.2 模块使用说明	3
2.3.1. SPI 通信	3
2.3.2. SCI 寄存器	5
2.3.3. 播放音频文件.....	7
3. 结构尺寸	8
4. 其他	8

1. 特性参数

特性	参数
接口特性	3.3V 工作
解码格式	WMA 4.0/4.1/7/8/9 5-384kbps 所有流文件、MP3 和 WAV 流
编码格式	IMA ADPCM 编码
板载录音	支持
存储	5.5KB 片上 RAM
DAC 分辨率	18 位
其他特性	音量控制、高低音控制

2. 使用说明

2.1 模块引脚说明



黑色弯排针管脚:

管脚名称	与 MCU 连接	说明
3.3V	3.3V	3.3V 供电
GND	GND	电源地
MISO	MOSI	SPI 总线数据输入线
MOSI	MISO	SPI 总线数据输出线
SCLK	SCK	SPI 总线时钟线
xCS	I/O1	片选输入 (低电平有效)
xDCS	I/O2	数据片选/字节同步
DREQ	I/O3	数据请求
xRSET	I/O4	复位引脚 (硬件复位, 低电平有效)

黄色排针:

管脚名称	说明
P0	通用 GPIO 0
P1	通用 GPIO 1
P2	通用 GPIO 2
P3	通用 GPIO 3
AGND	模拟地
OUT	左声道输出

如果在引导时间里, GPIO0 被上拉电阻拉到高电平, VS1003 尝试从外部 SPI 存储器中引导, SPI 引导重定义的管脚如下:

正常模式 SPI 引导模式

正常模式	SPI 引导模式
GPIO0	xCS
GPIO1	CLK
DREQ	MOSI
GPIO2	MISO

必须有 16 位地址 (即至少 1KB) 的 SPI 串行 EEPROM。SPI 时钟速度在 VS1003 工作在 12.288MHz 时为 245KHz。此存储器中的前三字节必须是 0x50,0x26,0x48。

2.2 模块使用说明

2.3.1. SPI 通信

VS1003B MP3 Board 模块通过 SPI 接口与外部控制器连接, VS1003 的控制以及音频数据都是通过 SPI 接口, VS1003 通过 7 根信号线共同控制器, 分别是: xRSET、XCS、XDCS、SI、SO、SCK 和 DREQ。其中 xRSET 是 VS1003 的复位控制线, 低电平有效。DREQ 是数据请

求线,用于通知控制器 VS1003 是否可以接收数据。SI(MOSI)、SO(MISO)、SCK 则是 VS1003 的 SPI 通信接口,他们在 XCS 和 XDCS 的控制下执行不同的数据通信。

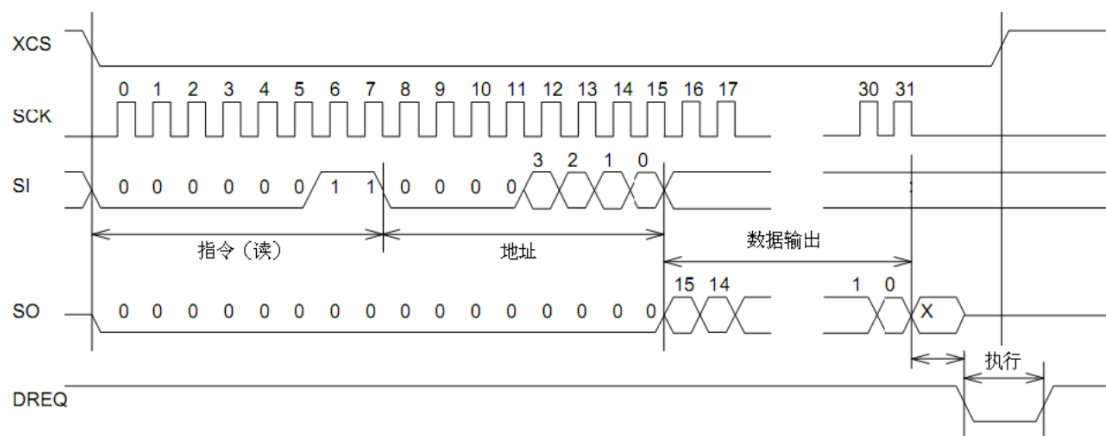
VS1003 的 SPI 支持两种模式: 1, VS1002 有效模式(即新模式)。2, VS1001 兼容模式。这里我们仅介绍 VS1002 有效模式(此模式也是 VS1003 的默认模式)。下表是在新模式下 VS1003 的 SPI 信号线功能描述:

SDI 管脚	SCI 管脚	描述
XDCS	XCS	低电平有效片选输入, 高电平强制使串行接口进入 standby 模式, 结束当前操作。SO 变成高阻态。如果 SM_SDISHARE 位设置为 1, 则不使用 XDCS, 而有 XCS 内部反相后代替 XDCS。不过不推荐这种设置方式。
SCK		串行时钟输入。SCK 在传输的时候可以被打断, 但是必须保持 XCS/XDCS 低电平不变, 否则传输将中断。
SI		串行数据输入。在片选有效的情况下, SI 在 SCK 的上升沿处采样。所以, MCU 必须在 SCK 的下降沿上更新数据
SO		串行输出, 在读操作时, 数据在 SCK 的下降沿处从此脚移出; 在写操作时为高阻态。

VS1003 的 SPI 数据传送, 分为 SDI 和 SCI, SDI 用来传输数据, SCI 用于传输命令。SDI 数据传输非常简单, 就是标准 SPI 通信, 不过 VS1003 的数据传输都是通过 DREQ 控制, 主机必须在判断 DREQ 有效(高电平有效)后, 才可以发送数据, 每次可以发送 32 字节。

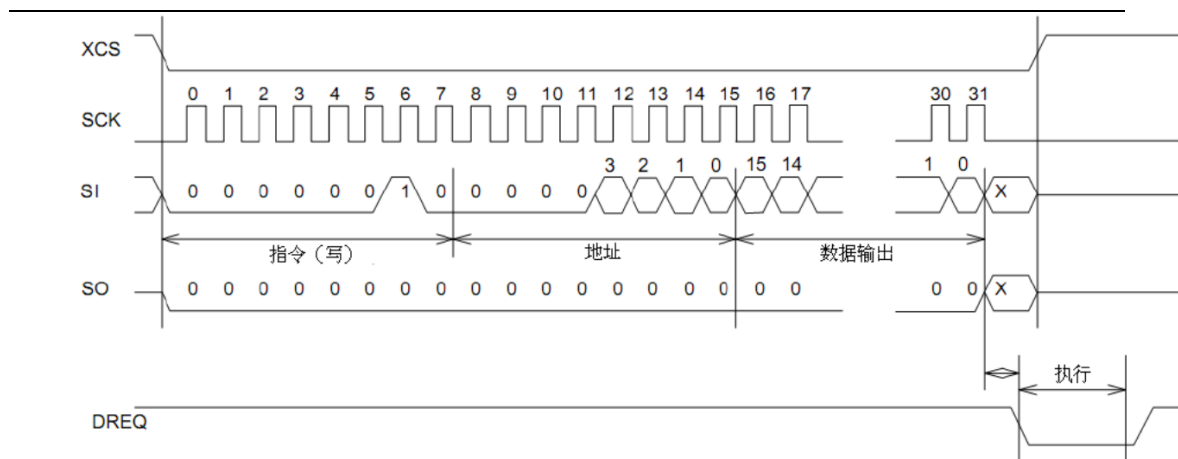
SCI 串行总线命令接口包含了一个指令字节、一个地址字节和一个 16 位的数据字。读写操作可以读写单个寄存器, 在 SCK 的上升沿读出数据位, 所以主机必须在下降沿刷新数据。SCI 的字节数据总是高位在前低位在后。第一个字节指令字节, 只有 2 个指令, 也就是读和写, 读指令为: 0X03, 写指令为: 0X02。

SCI 读时序如图所示:



从上图可以看出, 向 VS1003 读取数据, 通过先拉低 XCS, 然后发送读指令(0X03)再发送一个地址, 最后, 我们在 SO 线(MISO)上就可以读到输出的数据了。而同时 SI(MOSI)上的数据将被忽略。

SCI 的写时序如下图:



SCI 写时序和 SCI 读时序类似，都是先发指令，再发地址。不过写时序中，我们的指令是写指令（0X02），并且数据是通过 SI 写入 VS1003 的，SO 则一直维持低电平。在上两个图中，DREQ 信号上都产生了一个短暂的低脉冲，也就是执行时间。这段时间，是不允许外部打断的。

2.3.2. SCI 寄存器

VS1003 总共有 16 个 SCI 寄存器，通过这些寄存器实现 VS1003 的各种控制，VS1003 的所有 SCI 寄存器如下表所示：

SCI 寄存器				
寄存器	类型	复位值	缩写	描述
0x00	RW	0x0800	MODE	控制模式
0x01	RW	0X003C	STATUS	VS1003 状态
0x02	RW	0X0000	BASS	内置高低音增强器
0x03	RW	0X0000	CLOKF	时钟频率+倍频数
0x04	RW	0X0000	DECODE_TIME	解码时间
0x05	RW	0X0000	AUDATA	各种音频数据
0x06	RW	0X0000	WRAM	RAM 写/读
0x07	RW	0X0000	WRAMADDR	RAM 读写地址
0x08	R	0X0000	HDATA0	流头数据 0
0x09	R	0X0000	HDATA1	流头数据 1
0x0A	RW	0X0000	AIADDR	用户代码起始地址
0x0B	RW	0X0000	VOL	音量控制
0x0C	RW	0X0000	AICTRL0	应用控制寄存器 0
0x0D	RW	0X0000	AICTRL1	应用控制寄存器 1
0x0E	RW	0X0000	AICTRL2	应用控制寄存器 2
0X0F	RW	0X0000	AICTRL3	应用控制寄存器 3

以下介绍几个主要的寄存器：

MODE 寄存器：该寄存器用于控制 VS1003 的模式，改寄存器的初始值为 0x0800，MODE 的各位描述如下：

位	名称	功能	值	说明
0	SM_DIFF	差分	0	正常的相同音频
			1	左通道想相反
1	SM_LAYER12	允许 MPEG layers I&II 解码	0	不允许
			1	允许
2	SM_RESET	软件复位	0	不复位
			1	复位
3	SM_CANCEL	取消当前文件的解码	0	不取消
			1	取消
4	SM_EARSPEAKER_LO	EarSpeaker 低位设定	0	关闭
			1	激活
5	SM_TESTS	允许 SDI 测试	0	不允许
			1	允许
6	SM_STREAM	流模式	0	不是
			1	是
7	SM_EARSPEAKER_HI	EarSpeaker 高位设定	0	关闭
			1	激活
8	SM_DACT	DCLK 的有效边沿	0	上升沿
			1	下降沿
9	SM_SDIORD	SDI 位顺序	0	MSB 在前
			1	MSB 在后
10	SM_SDISHARE	SM_SDISHARE	0	不共享
			1	共享
11	SM_SDINew	VS1002 本地 SPI 模式	0	非本地模式
			1	本地模式 (新模式)
12	SM_ADPCM	ADPCM 录音激活	0	不激活
			1	激活
13	SM_ADPCM_HP	ADPCM 高通滤波允许	0	不允许
			1	允许
14	SM_LINE_IN	ADPCM 音源选择	0	麦克风
			1	线路输入
15	SM_CLK_RANGE	SM_CLK_RANGE	0	12...13Mhz
			1	24...26Mhz

这个寄存器，我们这里只关注第 2 位和第 11 位，也就是 **SM_RESET** 和 **SM_SDINew**。这里 **SM_RESET**，可以提供一次软复位，建议在每播放一首歌曲之后，软复位一次。**SM_SDINew** 为模式设置位，这里我们选择的是新模式，所以设置该位为 1 (默认的设置)。单纯播放音乐的时候，我们只需要设置这两个位就差不多了，其他位的设置，请参考 **VS1003** 的数据手册

BASS 寄存器，该寄存器可以用于设置 **VS1003** 的高低音效。该寄存器的各位描述如表所示：

名称	位	描述
ST_AMPLITUDE	15:12	高音控制，1.5dB 步进 (-8..7, 0 表示关闭)

ST_AMPLITUDE	11:8	最低频限 1000Hz 步进 (0..15)
SB_AMPLITUDE	7:4	低音加重, 1dB 步进 (0..15, 0 表示关闭)
SB_FREQLIMIT	3:0	最低频限 10Hz 步进 (2..15)

通过这个寄存器以上位的一些设置, 我们可以随意配置自己喜欢的音效 (其实就是高低音的调节)。

CLOCKF 寄存器, 这个寄存器用来设置时钟频率、倍频等相关信息, 该寄存器的各位描述如表所示:

CLOCKF 寄存器			
位	15:13	12:11	10:0
名称	SC_MULT	SC_ADD	SC_FREQ
描述	时钟倍频数	允许倍频	时钟频率
说明	$CLKI = XTALI \times (SC_MULT \times 0.5 + 1)$	倍频增量 $= SC_ADD * 0.5$	当外部时钟频率不为 12.288Mhz 时, 外部时钟的频率。外部时钟频率为 12.288Mhz 时, 此部分设置为 0

SC_FREQ 是以 4Khz 为步进的一个时钟寄存器, 当外部时钟不是 12.288M 的时候, 其计算公式为:

$$SC_FREQ = (XTALI - 8000000) / 4000$$

式中为 XTALI 的单位为 Hz。上表中 CLKI 是内部时钟频率, XTALI 是外部晶振的时钟频率。由于我们使用的是 12.288M 的晶振, 所以 SC_FREQ = 0;

DECODE_TIME 寄存器。该寄存器是一个十六位的寄存器, 用于存放解码时间, 以秒钟为单位, 我们通过读取该寄存器的值, 就可以得到解码时间了。不过它是一个累计时间, 所以我们需要在每首歌播放之前把它清空一下, 以得到这首歌的准确解码时间。

VOL 寄存器, 该寄存器用于控制 VS1003 的输出音量, 该寄存器可以分别控制左右声道的音量, 每个声道的控制范围为 0~254, 每个增量代表 0.5db 的衰减, 所以该值越小, 代表音量越大。比如设置为 0X0000 则音量最大, 而设置为 0XFEFE 则音量最小。注意: 如果设置 VOL 的值为 0XFFFF, 将使芯片进入掉电模式!

其他寄存器, 这里就不一一介绍了, 请大家参考 VS1003 的数据手册, 里面有每个寄存器的详细介绍。

2.3.3. 播放音频文件

用模块播放音频文件非常的简单, 一般的音频文件 (MP3/WMA/OGG/WAV/MIDI/AAC 等), 只需要简单的 3 步操作即可实现音频播放。

(1) 复位 VS1003

这里包括了硬复位和软复位, 是为了让 VS1003 的状态回到原始状态, 准备解码下一首歌曲。这里建议在每首歌曲播放之前都执行一次硬件复位和软件复位, 以便更好的播放音乐。

(2) 配置 VS1003 相关寄存器

配置的寄存器包括 VS1003 的模式寄存器 (MODE)、时钟寄存器 (CLOCKF)、音调寄存器 (BASS)、音量寄存器 (VOL) 等。

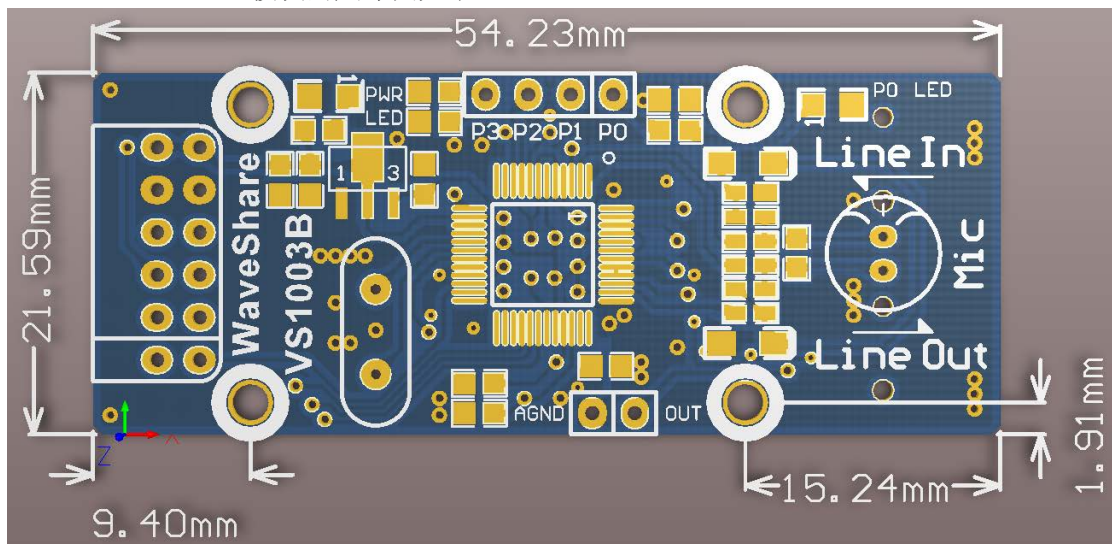
(3) 发送音频数据

当经过以上两步配置以后, 我们剩下来要做的事情, 就是往 VS1003 里面扔音频数据了,

只要是 VS1003 支持的音频格式, 直接往里面丢就可以了, VS1003 会自动识别, 并进行播放。不过发送数据要在 DREQ 信号的控制下有序的进行, 不能乱发。只要 DREQ 变高, 就向 VS1003 发送 32 个字节。然后继续等待 DREQ 变高, 直到音频数据发送完。

3. 结构尺寸

VS1003B MP3 Board 模块的尺寸图如下:



4. 其他

购买地址:

<http://www.waveshare.net/shop/Vs1003B-MP3-Board.htm>

资料地址:

http://www.waveshare.net/wiki/Vs1003B_MP3_Board