

# VS1003 - MP3/WMA 音频编解码器

## 特征

- 解码 MPEG 1 和 2 音频层 III (CBR + VBR + ABR); WMA 4.0/4.1/7/8/9 所有配置文件 (5-384kbit/s); WAV (PCM + IMA AD-相变材料) ; 通用 MIDI / SP-MIDI 文件
- 从麦克风或线路输入编码 IMA ADPCM
- 对 MP3 和 WAV 的流媒体支持
- 低音和高音控制
- 使用单个 12..13 MHz 时钟运行
- 内部 PLL 时钟倍频器
- 低功耗运行
- 高品质片上立体声 DAC, 通道间无相位误差
- 能够驱动 The i 的立体声耳机驱动器 30Ω 加载
- 模拟、数字多路独立工作电压意大利和 I/O
- 用于用户代码/数据的 5.5 KiB 片上 RAM
- 串行控制和数据接口
- 可用作从协处理器
- 用于特殊应用的 SPI 闪存启动
- 用于调试目的的 UART
- 可以通过软件和 4 个 GPIO 引脚添加新功能

## 描述

VS1003 是一款单芯片 MP3/WMA/MIDI 音频解码器和 ADPCM 编码器。它包含高性能、专有的低功耗 DSP 处理器

VSDSP 核心<sup>4</sup>, 工作数据存储器, 5 KiB

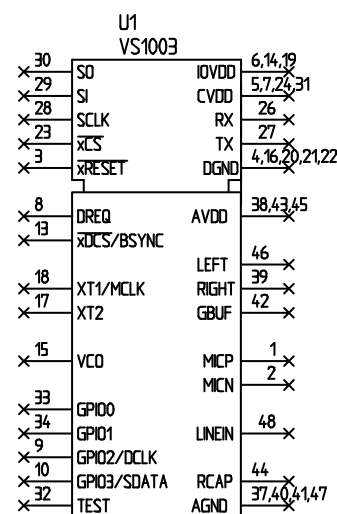
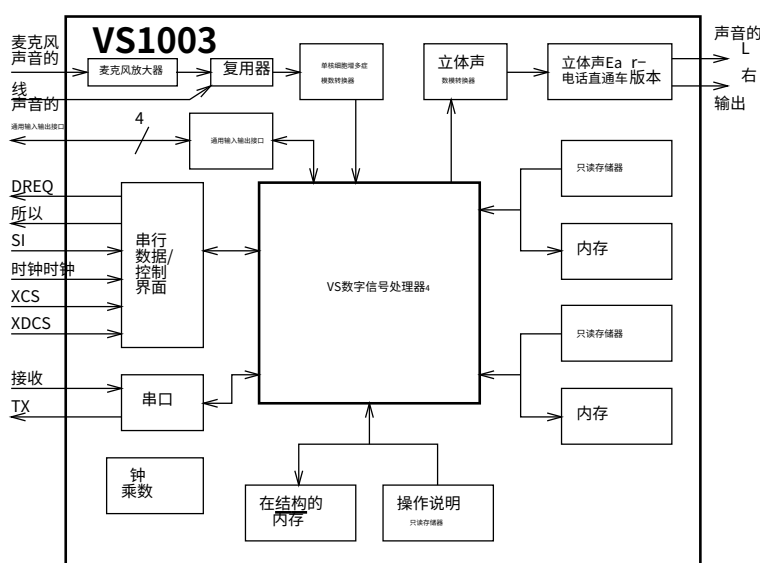
仪器 用于用户阳离子、串行控制和输入数据交互的应用程序 RAM 和 0.5 KiB 数据 RAM、4 个通用 I/O 面孔 引脚、一个 UART, 作为高质量可变采样率单出色通道

模数转换器 和立体声 DAC, 然后是耳机连接器和公共缓冲放大器。

VS1003 通过 seput 总线接收其输入比特流, 并作为里亚系统从机进行监听。输入流被解码并通过音量控制传递到 18 位过采样位 sigma-delta DAC。数字 解码通过串行控制总线进行。除了

巨魔

基本的解码后, 可以将应用程序功能 (例如 DSP 效果规格 果) 添加到用户 RAM 中。内存





## 内容

<b>1 许可证</b>	<b>9</b>
<b>2 免责声明</b>	<b>9</b>
<b>3 定义</b>	<b>9</b>
<b>4 特性与规格</b>	<b>10</b>
4.1 绝对最大额定值	10
4.2 推荐的操作条件。	10
4.3 模拟特性。	11
4.4 功耗。	12
4.5 数字特性。	12
4.6 开关特性-引导初始化。	12
4.7 典型特征。	13
4.7.1 线路输入 ADC。	13
4.7.2 麦克风输入ADC。	13
4.7.3 右输出和左输出。	14
<b>5 封装及引脚说明</b>	<b>15</b>
5.1 包。	15
5.1.1 LQFP-48。	15
5.1.2 BGA-49。	15
5.2 LQFP-48 和 BGA-49 引脚说明。	16
<b>6 连接图, LQFP-48</b>	<b>18</b>

<b>7</b>	<b>SPI总线</b>	<b>19</b>
7.1	概述	19
7.2	SPI 总线引脚说明。	19
7.2.1	VS1002 本机模式（新模式）	19
7.2.2	VS1001 兼容模式。	19
7.3	数据请求引脚 DREQ。	20
7.4	串行数据接口 (SDI) 的串行协议。	20
7.4.1	概述。	20
7.4.2	VS1002 本机模式下的 SDI（新模式）	20
7.4.3	VS1001 兼容模式下的 SDI。	21
7.4.4	被动SDI 模式。	21
7.5	串行命令接口 (SCI) 的串行协议。	21
7.5.1	概述。	21
7.5.2	SCI 阅读。	22
7.5.3	SCI 写入。	22
7.6	SPI 时序图。	23
7.7	设置 SMSDINEW 和 SMSDISHARED 的 SPL 示例。	24
7.7.1	两次 SCI 写入。	24
7.7.2	两个 SDI 字节。	24
7.7.3	两个 SDI 字节中间的 SCI 操作。	25
<b>8</b>	<b>功能说明</b>	<b>26</b>
8.1	主要特点。	26
8.2	支持的音频编解码器。	26
8.2.1	支持的 MP3（MPEG 第三层）格式。	26

8.2.2 支持的 WMA 格式。	27
8.2.3 支持的 RIFF WAV 格式。	28
8.2.4 支持的 MIDI 格式。	29
8.3 VS1003 的数据流。	30
8.4 串行数据接口 (SDI)	30
8.5 串行控制接口 (SCI)	31
8.6 SCI 寄存器。	31
8.6.1 SCI 模式 (RW)	32
8.6.2 SCISTATUS (RW)。	34
8.6.3 SCIBASS (RW)	34
8.6.4 SCICLOCKF (RW)。	35
8.6.5 SCIDE 代码时间 (RW)	36
8.6.6 SCIAUDATA (RW)	36
8.6.7 SCIWRAM (RW)	36
8.6.8 SCIWRAMADDR (W)。	36
8.6.9 SCIHDAT0 和 SCIHDAT1 (R)。	37
8.6.10 SCIAIADDR (RW)	38
8.6.11 SCIVOL (RW)。	39
8.6.12 SCIAICIRL[x] (RW)	39
<b>9 手术</b>	<b>40</b>
9.1 时钟。	40
9.2 硬件复位。	40
9.3 软件复位。	40
9.4 ADPCM 录音。	41

9.4.1 激活ADPCM 模式。	41
9.4.2 读取IMA ADPCM 数据。	41
9.4.3 添加 RIFF 标头。	42
9.4.4 播放ADPCM 数据。	43
9.4.5 采样率注意事项。	43
9.4.6 示例代码。	43
9.5 SPI 启动。	45
9.6 播放/解码。	45
9.7 馈送 PCM 数据。	45
9.8 SDI 测试。	46
9.8.1 正弦测试。	46
9.8.2 引脚测试。	46
9.8.3 内存测试。	47
9.8.4 SCI 测试。	47
<b>10 个 VS1003 寄存器</b>	<b>48</b>
10.1 谁需要阅读本章	48
10.2 处理器核心。	48
10.3 VS1003 内存映射。	48
10.4 SCI 寄存器。	48
10.5 串行数据寄存器。	48
10.6 DAC 寄存器。	49
10.7 GPIO 寄存器。	50
10.8 中断寄存器。	51
10.9 A/D 调制器寄存器。	52

1.04版, 2009-02-03







## 1 许可证

MPEG Layer-3 音频解码技术由 Fraunhofer IIS 和 Thomson 授权。

VS1003 包含微软的 WMA 解码技术。

**本产品受 Microsoft 的某些知识产权保护，未经 Microsoft 许可不得使用或进一步分发。**

## 2 免责声明

所有属性和数据均可能发生变化。

## 3 定义

**乙** 字节，8 位。

**乙** 少量。

**基** “吉比”  $=2^{10}=1024$  (IEC 60027-2) 。

**米** “梅比”  $=2^{20}=1048576$  (IEC 60027-2) 。

**数字信号处理器** VLSI 解决方案的 DSP 核心。

**瓦** 单词。在 VS DSP 中，指令字是 32 位宽，数据字是 16 位宽。



4 特性与规格

4.1 绝对最大额定值

范围	象征	最小	最大限度	单元
模拟正电源	电压源	- 0.3	2.85	V
数字正电源	颈椎病	- 0.3	2.7	V
I/O 正电源	IO电压	- 0.3	3.6	V
任何数字输出的电流			$\pm 50$	嘛
任何数字输入的电压		- 0.3	IOVDD+0.3 <sub>1</sub>	V
工作温度		- 40	+ 85	℃
储存温度		- 65	+ 150	℃

<sub>1</sub>不得超过 3.6V

4.2 推荐工作条件

范围	象征	最小	典型值	最大限度	单元
工作环境温度	e	- 40		+ 85	℃
模拟和数字地 <sub>1</sub>	AGND DGND		0.0		V
正模拟	电压源	2.6	2.8	2.85	V
正数	颈椎病	2.4	2.5	2.7	V
输入/输出电压	IO电压	CVDD-0.6V	2.8	3.6	V
输入时钟频率 <sub>2y</sub>	西塔利	12	12.288	13	兆赫兹
内部时钟频率	时钟信号	12	36.864	52.0 <sub>4</sub>	兆赫兹
内部时钟倍频器 <sub>3</sub>		1.0 ×	3.0 ×	4.5 × <sub>4</sub>	
主时钟占空比		40	50	60	%

<sub>1</sub>必须尽可能靠近器件连接在一起，以防止闩锁效应。

<sub>2</sub>以正确速度播放的最大采样率为 XTALI/256。

因此，XTALI 必须至少为 12.288 MHz 才能以正确的速度播放 48 kHz。

<sub>3</sub>复位值为1.0 ×。推荐SC 倍数=3.0 ×, SC 添加=1.0 × (SCL时钟F=0x9000)。

<sub>4</sub>52.0 MHz 是整个 CVDD 范围的最大时钟。(4.0 ×12.288 MHz=49.152 MHz 或4.0 ×13.0MHz=52.0 MHz)



4.3 模拟特性

除非另有说明A:VDD=2.85V,CVDD=2.5V, IOVDD=-2.8V, TA=-25..+70℃, XTALI=12.288MHz, DAC 使用 1307.894 Hz 满量程输出正弦波进行测试, 测量带宽 20..20000 Hz, 模拟输出负载: LEFT 至 GBUFΩ3,0右至 GBUF 30Ω。麦克风测试幅度 50 mVpp, f=1 kHz, 线路输入测试幅度 2.2 Vpp, f=1 kHz。

范围	象征	最小	典型值	最大限摩元
数模转换器分辨率			18	位
总谐波失真	总谐波失真		0.1	0.3 %
动态范围 (DAC 未静音, A 加权)	印尼盾		>90	D b
信噪比 (满量程信号)	信噪比	70 <sub>5</sub>	83 <sub>4</sub>	D b
通道间隔离 (串扰)		50	75	D b
通道间隔离 (串扰), 带 GB	UF		40	D b
通道间增益不匹配		- 0.5	±0.2	0.5 D b
频率响应		- 0.1		0.1 D b
满量程输出电压 (峰峰值)		1.3	1.5 <sub>1</sub>	1.7 脉压
线性相位偏差				5 °
模拟输出负载电阻	AOLR	16	30 <sub>2</sub>	Ω
模拟输出负载电容				100 PF
麦克风输入放大器增益	微细血管造影		26	D b
麦克风输入幅度			50	140 <sub>3</sub> 交流毫伏峰值
麦克风总谐波失真	MTHD		0.02	0.10 %
麦克风信噪比	MSNR	50 <sub>5</sub>	68	D b
线路输入幅度			2200	2800 <sub>3</sub> 交流毫伏峰值
线路输入总谐波失真	LTHD		0.015	0.10 %
线路输入信噪比	LSNR	60 <sub>5</sub>	86	D b
线路和麦克风输入阻抗			100	kΩ

典型值是对批次 4234011、周代码 0452 的约 5000 个器件进行测量的。

1通过 + 到 + 接线可实现 3.0 伏电压, 以实现单声道差分声音。

2AOLR 可能要低得多, 但比时间 w 典型的失真性能可能会受到影响。

3高于典型幅度, 谐波失真会增加。

4未加权时, A 加权大约好 3 dB。

5由于生产测试仪的噪音水平而限制较低。



4.4 功耗

使用 MPEG 1.0 Layer-3 128 kbit/s 样本进行测试并生成正弦波。全音量输出。XTALI 12.288 MHz。内部时钟倍频3r。0×。CVDD = 2.5V，AVDD = 2.8V。

范围	最小	典型值	最大限度	单元
电源消耗 AVDD、复位		0.6	5.0	μA
电源消耗 CVDD、复位、+	25	3.7	40.0	μA
电源消耗 CVDD、复位、+	85		200.0	μA
电源消耗AVDD，正弦测试，Ω30	+ GBUF	36.9		嘛
电源消耗CVDD，正弦测试		12.4		嘛
电源消耗 AVDD，空载		7.0		嘛
电源消耗 AVDD、输出负载Ω30		10.9		嘛
电源消耗 AVDD, 3Ω0+GBUF		16.1		嘛
电源消耗 CVDD		17.5		嘛

4.5 数字特性

范围	象征	最小	典型值	最大限度	单元
高电平输入电压		0。7×IO电压		IOVDD+0.3 <sub>1</sub>	V
低电平输入电压		- 0.2		0。3×IO电压	V
高电平输出电压 <sub>我</sub> = -1.0 毫安		0。7×IO电压			V
低电平输出电压 <sub>我</sub> = 1.0毫安				0。3×IO电压	V
输入漏电流		- 1.0		1.0	μA
SPI 输入时钟频率 <sub>2y</sub>				<sub>时钟信号</sub> 7	兆赫兹
所有输出引脚的上升时间，负载 = 50 pF				50	纳秒

<sub>1</sub>不得超过 3.6V  
<sub>2</sub>SCI 阅读的价值。SCI 和 SDI 写入  $\frac{CWLKI}{4}$ 。

4.6 开关特性 - 引导初始化

范围	象征	最小	最大限度	单元
XRESET有效时间		2		西塔利
XRESET 对软件读取无效	y	16600	50000 <sub>1</sub>	西塔利
上电复位、CVD 上升时间	D	10		电压/秒

<sub>1</sub>初始化完成后，DREQ 上升。在此之前您不应发送任何数据或命令。

## 4.7 典型特性

### 4.7.1 线路输入 ADC

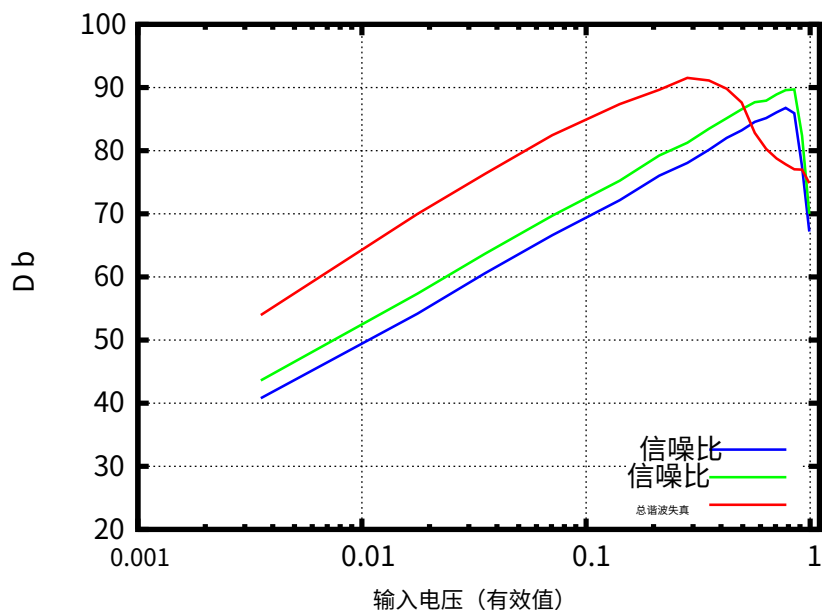


图 1：测得的 LINEIN 引脚的 ADC 性能。X 轴是 1 kHz 正弦输入的均方根幅度。曲线为未加权信噪比（蓝色）、A 加权信噪比（绿色）和未加权信噪比（红色）。ADC 采样率为 48 kHz（主时钟 12.288 MHz），噪声计算范围为 0 至 20 kHz。

### 4.7.2 麦克风输入 ADC

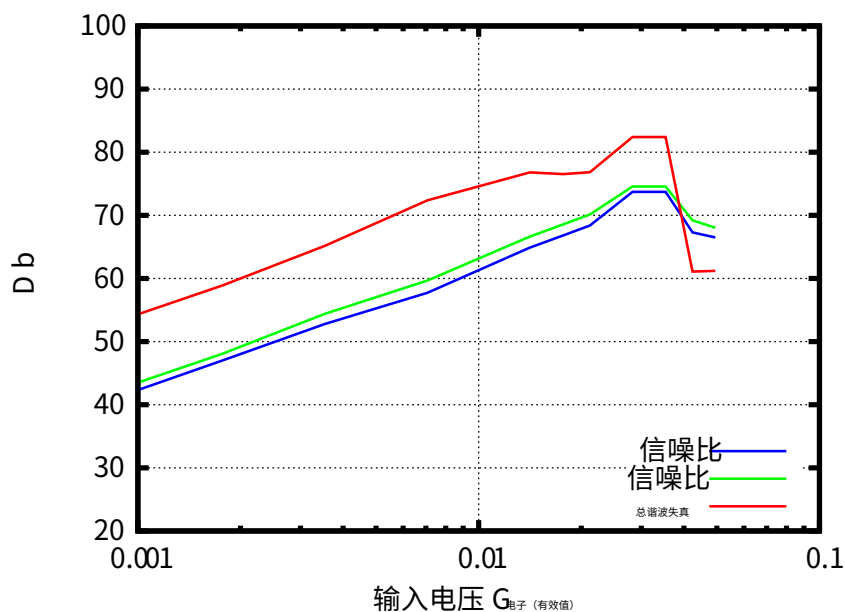


图 2：测量的 AD C 表现 MIC pi ns（微分）。其他塞蒂 ngs与图1相同。

### 4.7.3 右输出和左输出

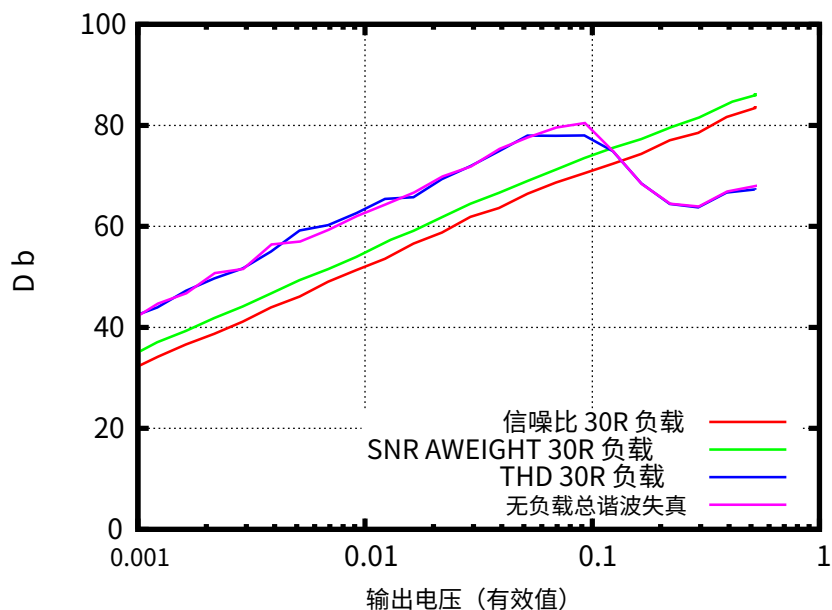


图 3：使用 1 kHz 生成的正弦波测得的右（或左）输出性能。DAC 采样率为 48 kHz（主时钟 12.288 MHz），噪声计算范围为 0 至 20 kHz。

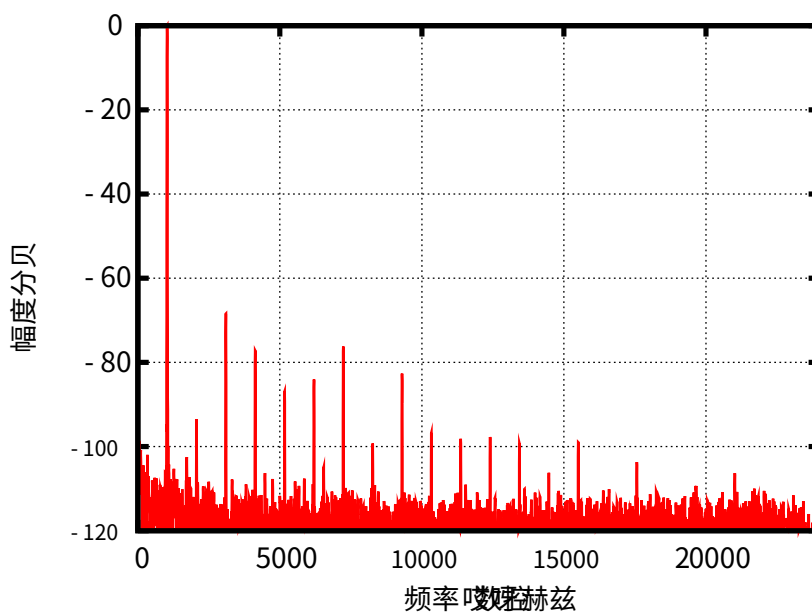


图 4：典型 A 光谱。米 RI 的 GHT（或左）o 呢 Pt 与格言 你米级和 30 欧姆负载。设置是一样的是我 n 图 3。

## 5 封装及引脚说明

### 5.1 封装

LPQFP-48 和 BGA-49 均不含铅 (Pb)，且符合 RoHS 标准。RoHS 是简称关于限制在电气和电子设备中使用某些有害物质的指令 2002/95/EC。恩特

#### 5.1.1 LQFP-48

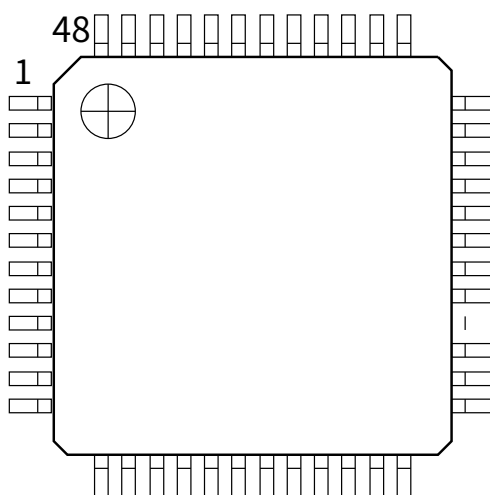
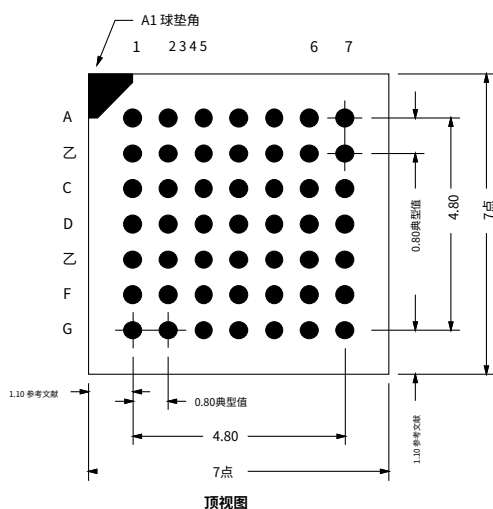


图 5：引脚配置，LQFP-48。

LQFP-48 封装尺寸为 HA

TTp://www.vlsi.fi/

#### 5.1.2 BGA-49



数字 6：引脚 C 配置，BGA-49。

BGA-49 封装尺寸为 HA TT

p://www.vlsi.fi/

## 5.2 LQFP-48 和 BGA-49 引脚说明

引脚名称	LQFP-48 针	球栅阵列49球	别针类型	功能
微量ICP	1	C3	人工智能	正差分麦克风输入，自偏置
麦克风网络	2	C2	人工智能	负差分麦克风输入，自偏置
重置	3	B1	DI	低电平有效异步复位
地线0	4	D2	地线	核心和 I/O 接地
电压源0	5	C1	CPWR	核心电源
IOVDD0	6	D3	IOPWR	输入输出电源
电压源1	7	D1	CPWR	核心电源
DREQ	8	E2	做	数据请求，输入总线
GPIO2/DCLK <sub>1</sub>	9	E1	迪奥	通用 IO 2/串行输入数据总线时钟
GPIO3/SDATA <sub>1</sub>	10	F2	迪奥	通用IO 3 /串行数据输入
XDCS/BSYNC <sub>1</sub>	13	E3	DI	数据片选/字节同步
IOVDD1	14	F3	IOPWR	输入输出电源
压控振荡器	15	G2	做	仅用于测试（时钟 VCO 输出）
地线1	16	F4	地线	核心和 I/O 接地
XTALO	17 号	G3	AO	晶体输出
西塔利	18	E4	人工智能	晶振输入
IOVDD2	19	G4	IOPWR	输入输出电源
IOVDD3		F5	IOPWR	输入输出电源
地线2	20		地线	核心和 I/O 接地
地线3	21	G5	地线	核心和 I/O 接地
地线4	22	F6	地线	核心和 I/O 接地
XCS	23	G6	DI	片选输入（低电平有效）
电压源2	24	G7	CPWR	核心电源
接收	26	E6	DI	UART接收，不使用则连接IOVDD
TX	27	F7	做	串口发送
时钟时钟	28	D6	DI	串行总线时钟
SI	29	E7	DI	串行输入
所以	30	D5	溶解氧	串行输出
CVDD3	31	D7	CPWR	核心电源
测试	32	C6	DI	保留用于测试，连接到 IOVDD
GPIO0 / SPIBOOT <sub>1</sub>	33	C7	迪奥	通用IO 0 / SPIBOOT，使用10Ω0pKull-down电阻 <sub>2r</sub>
GPIO1	34	B6	迪奥	通用IO 1
AGND0	37	C5	APWR	模拟地、低噪声参考
电压源0	38	B5	APWR	模拟电源
正确的	39	A6	AO	右声道输出
AGND1	40	B4	APWR	模拟地
AGND2	41	A5	APWR	模拟地
GBUF	42	C4	AO	耳机常用缓冲器
电压源1	43	A4	APWR	模拟电源
RCAP	44	B3	一体机	滤波电容参考
电压源2	45	A3	APWR	模拟电源
左边	46	B2	AO	左声道输出
AGND3	47	A2	APWR	模拟地
线路输入	48	A1	人工智能	线路输入

<sup>1</sup>第一个引脚功能在新模式下处于活动状态，后者在兼容模式下处于活动状态。

<sup>2</sup>除非使用下拉电阻，否则会尝试 SPI Boot。详细信息请参见第 9.5 章。





引脚类型:

类型	描述	类型	描述
DI	数字输入, CMOS 输入焊盘	AO	模拟输出
做	数字输出, CMOS 输入焊盘	一体机	模拟输入/输出
迪奥	数字输入/输出	APWR	模拟电源引脚
溶解氧	数字输出, CMOS 三态输出 P	地线	核心或 I/O 接地引脚
人工智能	模拟输入	CPWR	核心电源引脚
		IOPWR	I/O电源引脚

在 BGA-49 中, 无连接球有 A7、B7、D4、E5、F1、G1。在 LQFP-48 中, 无连接引脚为 11、12、25、35、36。

## 6 连接图, LQFP-48

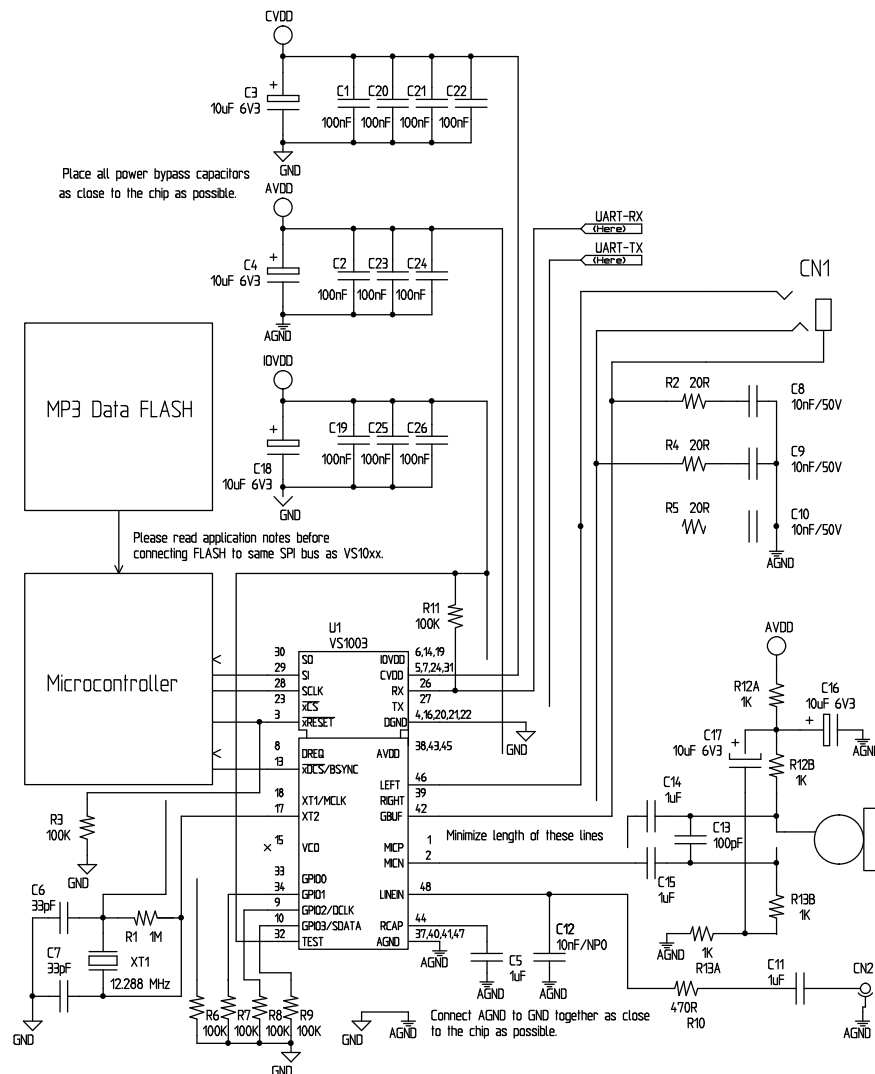


图 7: 使用 LQFP-48 的典型连接图。

通讯缓冲器GBUF可用于耳机的公共电压 (1.24 V)。这将消除 VS1003 需要大隔离电容线路输出。因此音频输出引脚可以直接连接到耳机连接中的 s 可能器。

GBUF 在任何情况下都不得接地。如果GBUF是RIGHT必须提供耦合电容。为了保持 未使用, LEFT 和 uld GBUF 稳定, 你必须抵抗托尔和 电容器e文什 未使用 GBUF。详细信息请参见应用笔记始终带有 Is。

乌努斯编辑GPI O 针守我有一个 pul l-向下电阻。

如果UART 为 n 未使用, RX应该 连接到 IOVDDA nd TX 未连接。

请勿连接 等任何外部XTAL 的实际负载 奥。

注: 这个C 连接a假设 SSMDINew i s 处于活动状态 (参见 C哈 第 8.6.1 节)。如果还 中 是 SS用d、xDCS 应绑在低处或高处 (参见第 7.2.1 章)。

7 SPI总线

7.1 概述

SPI 总线最初用于某些 Motorola 设备，现已用于 VS1003 的串行数据接口 SDI（第 7.4 和 8.4 章）和串行控制接口 SCI（第 7.5 和 8.5 章）。

7.2 SPI 总线引脚说明

7.2.1 VS1002 本机模式（新模式）

当 SD 时，这些模式在 VS1003 上有效。MINEW 设置为 1（启动时默认）。DCLK 和 SDATA 不用于数据传输，可用作通用 I/O 引脚（GPIO2 和 GPIO3）。BSYNC 功能更改为数据接口片选（XDCS）。

SDI 引脚	SCI 引脚	描述
XDCS	XCS	低电平有效片选输入。高电平强制串行接口进入待机模式，结束当前操作。高电平还会迫使 s 输出 (SO) 进入高阻抗状态。如果 SSMDISHARE 为 1，则引脚不使用 XDCS，但信号是通过反转 XCS 在内部生成的。
SCK		串行时钟输入。串行时钟还在内部用作寄存器接口的主时钟。  SCK 可以是门控的或连续的。无论哪种情况，XCS 变低后的第一个上升沿时钟 e 标记要写入的第一个位。
SI		串行输入。如果片选处于活动状态，则在 CLK e 上升沿对 SI 进行采样。
-	所以	串行输出。在读取过程中，数据在 SCK 下降沿移出。在写入时 SO 处于高阻抗状态。

7.2.2 VS1001 兼容模式

SM 时此模式有效。SDINEW 设置为 0。在此模式下，DCLK、SDATA 和 BSYNC 有效。

SDI 引脚	SCI 引脚	描述
-	XCS	低电平有效片选输入。高电平强制串行接口进入待机模式，结束当前操作。高电平还会迫使输出 (SO) 进入高阻抗状态。
同步同步	-	SDI 数据与 BSYNC 的上升沿同步。
时钟脉冲	SCK	串行时钟输入。串行时钟还在内部用作寄存器接口的主时钟。  SCK 可以是门控的或连续的。无论哪种情况，XCS 变低后的第一个上升沿时钟都标记要写入的第一个位。
数据传输	SI	串行输入。如果 XCS 为低电平，则在 SCK 上升沿对 SI 进行采样。
-	所以	串行输出。在读取过程中，数据在 SCK 下降沿移出。在写入时 SO 处于高阻抗状态。

### 7.3 数据请求引脚 DREQ

DREQ 引脚/信号用于指示 VS1003 的 FIFO 是否能够接收数据。如果 DREQ 为高电平，VS1003 可以获取至少 32 字节的 SDI 数据或一个 SCI 命令。当不满足这些标准时，DREQ 变低，发送方应停止传输新数据。

由于有 32 字节的安全区域，发送器一次最多可以发送 32 字节的 SDI 数据，而无需检查 DREQ 的状态，从而使低速微控制器更容易控制 VS1003。

注意：DREQ 可能随时变低或变高，即使在字节传输期间也是如此。因此，DREQ 只能用于决定是否发送更多字节。它不应该中止已经开始的传输。

注意：在 VS10XX 产品直至 VS1002 中，DREQ 仅用于 SDI。在 VS1003 中 DREQ 也用来告诉 SCI 的状态。

在某些情况下，当 DREQ 为低电平时，您仍然想发送 SCI 命令。由于 DREQ 在 SDI 和 SCI 之间共享，因此如果 SDI 未准备好接收，则无法确定 SCI 命令是否已执行。在这种情况下，您需要在每个 SCI 命令之后有足够的延迟，以确保不会错过任何一个命令。第 8.6 节中的 SCI 寄存器表给出了每个 SCI 寄存器写入的最坏情况处理时间。

### 7.4 串行数据接口 (SDI) 的串行协议

#### 7.4.1 概述

串行数据接口工作在从机模式下，因此 DCLK 信号必须由外部电路产生。

数据 (SDATA 信号) 可以在 DCLK 的上升沿或下降沿输入 (第 8.6 章)。

VS1003 假设其数据输入是字节同步的。SDI 字节可以先传输 MSb 或 LSb，具体取决于 SC 的内容 \_MIODE (第 8.6.1 章)。

固件能够接受 SDI 支持的最大比特率。

#### 7.4.2 VS1002 本机模式下的 SDI (新模式)

在 VS1002 本机模式下 (SM\_NEWMODE 为 1)，字节同步通过 XDCS 实现。的状态当数据字节传输正在进行时，XDCS 可能不会改变。为了始终保持数据同步，即使使用 VS1003 的板卡可能出现故障，建议时不时地打开 XDCS，例如在每个闪存数据块或几千字节之后打开一次，以确保主机和 VS1003 处于同步状态。同步中。

如果 SM\_SDISHARE 为 1，则 XDCS 信号是通过反转 XCS 输入在内部生成的。

对于新设计，建议使用 VS1002 本机模式。

7.4.3 VS1001 兼容模式下的 SDI

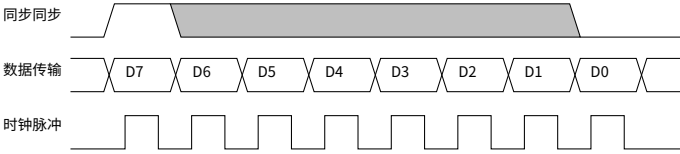


图 8：BSYNC 信号 - 一字节传输。

当VS1003在VS1001兼容模式下运行时，必须生成BSYNC信号以确保输入比特流的正确位对齐。第一个 DCLK 采样边沿（上升或下降，取决于所选极性），在此期间 BSYNC 为高电平，标记字节的第一位（LSB，如果使用 LSB 优先顺序，MSB，如果使用 MSB 优先顺序）。如果接收到最后一位时 BSYNC 为“1”，则接收器保持活动状态并且还会接收接下来的 8 位。

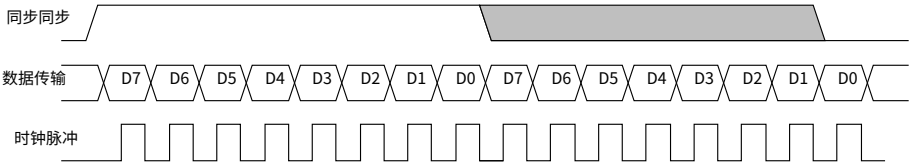


图 9：BSYNC 信号 - 两个字节传输。

7.4.4 被动SDI模式

如果 SM NEWMODE 为 0 并且 SMSDISHARE 为 1，则操作在其他方面与 VS1001 兼容模式类似，但仅在 BSYNC 信号为“1”时接收位。BSYNC 的上升沿仍用于同步。

7.5 串行命令接口（SCI）的串行协议

7.5.1 概述

串行命令接口 SCI 的串行总线协议（第 8.5 章）由指令字节、地址字节和一个 16 位数据字组成。每个读或写操作可以读或写一个寄存器。数据位在上升沿读取，因此用户应在下降沿更新数据。字节始终先发送 MSb。XCS 在整个操作期间应保持低电平，但如果需要，您可以在位之间暂停。

该操作由 8 位指令操作码指定。支持的指令是读和写。见下表。

操作说明		
姓名	操作码	手术
读	0b0000 0011	读取数据
写	0b0000 0010	写入数据

注意：VS1003 在每次 SCI 操作后将 DREQ 设置为低电平。持续时间取决于操作。在 DREQ 再次变高之前，不允许启动新的 SCI/SDI 操作。

### 7.5.2 SCI 读取

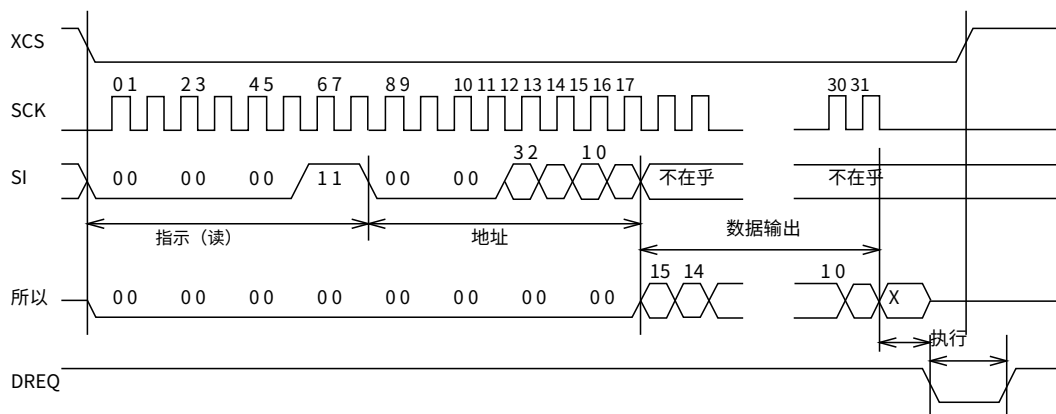


图 10: SCI 字读取

使用以下顺序读取 VS1003 寄存器，如图 10 所示。首先，将 XCS 线拉低以选择器件。然后，通过 SI 线传输 READ 操作码 (0x3)，后跟 8 位字地址。读入地址后，芯片将忽略 SI 上的任何其他数据。与接收到的地址相对应的 16 位数据将被移出到 SO 线上。

数据移出后，XCS 应被驱动为高电平。

当芯片进行读操作时，DREQ 会被短暂拉低。这是一个非常短的时间，不需要用户特别注意。

### 7.5.3 SCI 写入

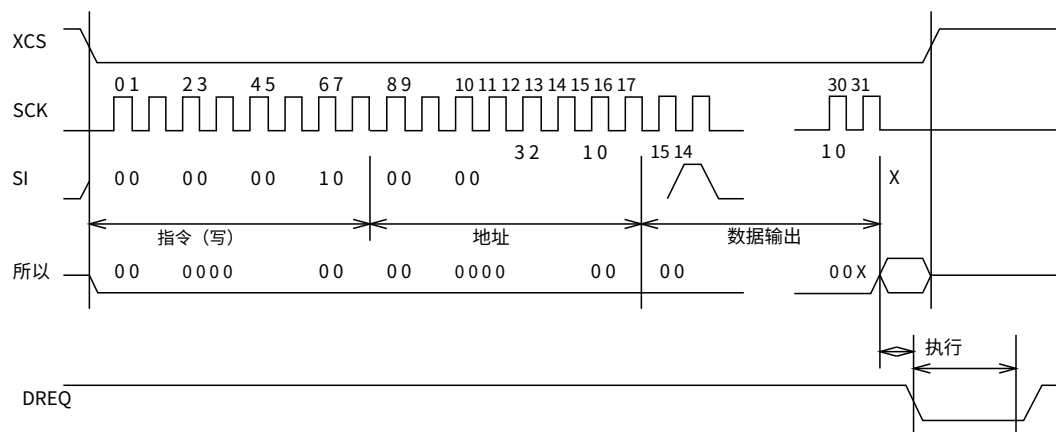


图 11: SCI 字写入

VS1003 寄存器我 sters 写自你 唱tH e佛 洛文G 后续编码 e、A s嘘 噢 被拉低以选择设备。然后WRITE操作码 (0x2) 被反式 通过 8 位字地址。

n 如图 11 所示。首先，XCS 线 米通过 SI 线进行了跟踪

在字移入并发送最后一个时钟后，XCS 应拉高以结束写序列。

发送最后一位后，DREQ 在寄存器更新期间被驱动为低电平，在图中标记为“执行”。该时间根据寄存器及其内容而变化（详细信息请参见第 8.6 章中的表格）。如果最大时间长于微控制器馈送下一个 SCI 命令或 SDI 字节所需的时间，则不允许在 DREQ 再次上升之前完成新的 SCI/SDI 操作。

7.6 SPI 时序图

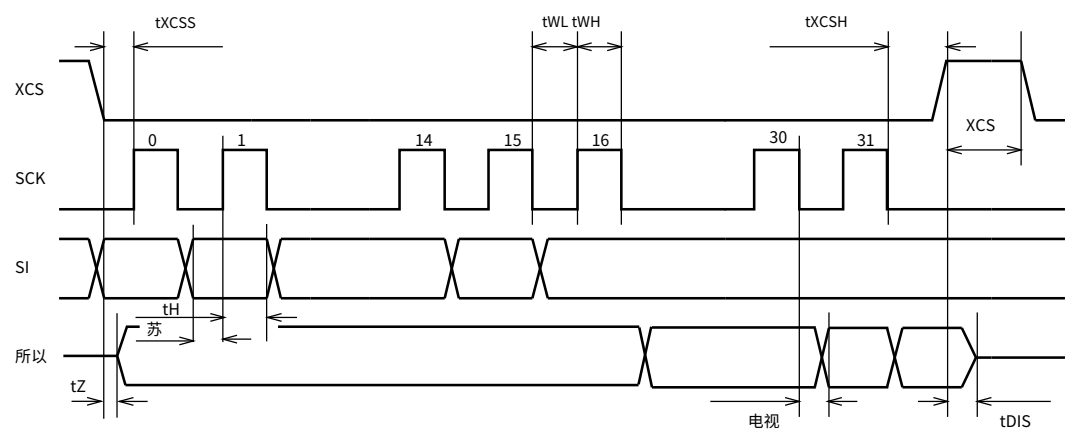


图 12：SPI 时序图。

象征	最小	最大限制单元
tXCSS	5	纳秒
苏	0	纳秒
tH	2	CLKI 周期
tZ	0	纳秒
tWL	2	CLKI 周期
总宽度	2	CLKI 周期
电视	2 (+25纳秒i)	CLKI 周期
tXCSH	1	时钟信号
XCS	2	CLKI 周期
tDIS		10 纳秒

125ns 是引脚加载 100pF 电容时的时间。电容越小，时间越短。

注：由于 tWL 和 tWH 以及 tH 需要至少 2 个时钟周期，因此可以轻松与交流时钟一起使用的 SPI 总线的最大速度是 VS1003 的 1/7 顺 赫罗 内部cl 锁定速度 CLKI。

注：虽然时序为导出模式，因此 d F哦么系统内部时钟CLKI 阿尔瓦开始1你。0p ×在 CLK= 我XTALI。

7.7 设置 SMSDINEW 和 SM SDISHARED 的 SPI 示例

7.7.1 两次 SCI 写入

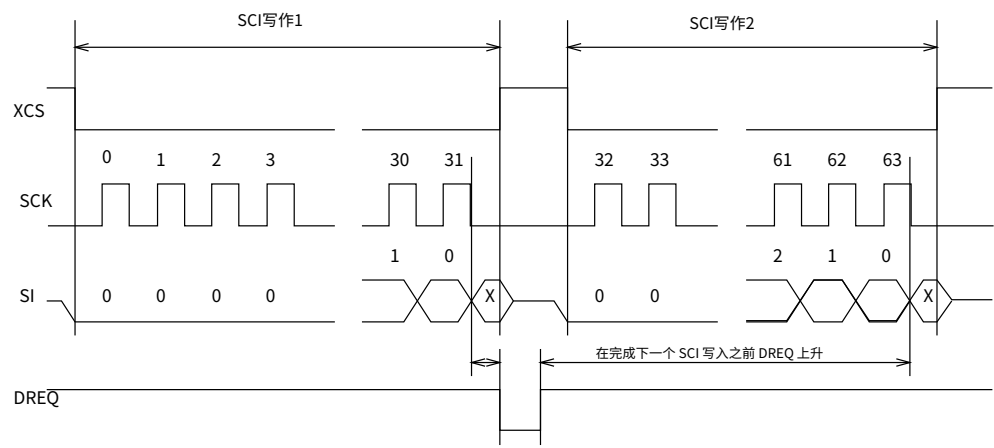


图 13：两个 SCI 操作。

图 13 显示了两个连续的 SCI 操作。注意米 x 我们 St 之间被提升到非活动状态 写道。如图所示，还必须遵守 DREQ。

7.7.2 两个 SDI 字节

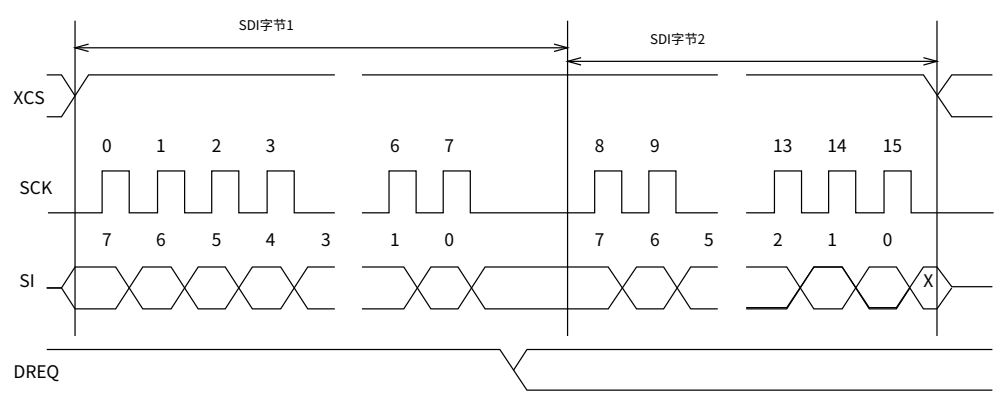


图 14：两个 SDI 字节。

SDI 数据与提升 e 同步 dx 的几何 CS 如图 14 所示。然而，每个字节 e 没有 需要单独同步。



### 7.7.3 两个 SDI 字节中间的 SCI 操作

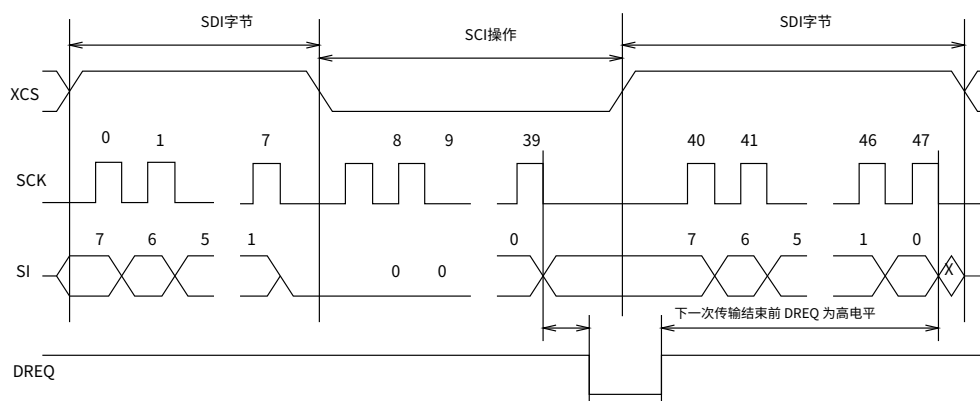


图 15：由 SCI 操作分隔的两个 SDI 字节。

图 15 显示了 SCI 操作如何嵌入到 SDI 操作之间。xCS 边沿用于同步 SDI 和 SCI。请记住尊重 DREQ，如图所示。

## 8 功能说明

### 8.1 主要特点

VS1003基于专有的数字信号处理器DV8000。它包含所有代码和数据MP3、WMA 和 WAV PCM + ADPCM 音频解码、MIDI 合成器以及串行接口、多速率立体声音频 DAC 以及模拟输出放大器和滤波器所需的内存。使用麦克风放大器和 A/D 转换器还支持 AD-PCM 音频编码。提供 UART 用于调试目的。

### 8.2 支持的音频编解码器

惯例	
标记	描述
+	支持格式
-	格式存在但不支持
	格式不存在

#### 8.2.1 支持的 MP3（MPEG 第三层）格式

MPEG 1.0<sub>1</sub>:

采样率/Hz	比特率/kbit/s													
	32	40	48	56	64	80	96	112	128	160	192	224	256	320
48000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
44100	+	+	+	+	+	+	+	+	+	+	+	+	+	+
32000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.0<sub>1</sub>:

采样率/Hz	比特率/kbit/s													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
24000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
22050	+	+	+	+	+	+	+	+	+	+	+	+	+	+
16000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

MPEG 2.5<sub>12</sub>:

采样率/Hz	比特率/kbit/s													
	8	16	24	32	40	48	56	64	80	96	112	128	144	160
12000	+	+	+	+	+	+	+	+	+	+	+	+	+	+
11025	+	+	+	+	+	+	+	+	+	+	+	+	+	+
8000	+	+	+	+	+	+	+	+	+	+	+	+	+	+

<sub>1</sub>还支持所有可变比特率 (VBR) 格式。

<sub>2</sub>由于 MPEG 2.5 不是标准格式，因此可能会出现不兼容性。

## 8.2.2 支持的 WMA 格式

支持 Windows Media Audio 编解码器版本 2、7、8 和 9。支持所有 WMA 配置文件（L1、L2 和 L3）。以前，流被分为 1、2a、2b 和 3 类。不支持 WMA 9 Professional 和 WMA 9 Lossless。该解码器已通过微软的一致性测试计划。

WMA 4.0 / 4.1:

采样率	比特率/kbit/s																
/赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+	+	+	+	+	+					
44100									+		+	+	+	+	+	+	
48000															+	+	

WMA 7:

采样率	比特率/kbit/s																
/赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+		+	+	+						
44100									+		+	+	+	+	+	+	+
48000															+	+	

WMA 8:

采样率	比特率/kbit/s																
/赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192
8000	+	+	+		+												
11025			+	+													
16000				+	+	+	+										
22050						+	+	+	+								
32000							+		+	+	+						
44100									+		+	+	+	+	+	+	+
48000															+	+	+

WMA 9:

采样率	比特率/kbit/s																			
/赫兹	5	6	8	10	12	16	20	22	32	40	48	64	80	96	128	160	192	256	320	
8000	+	+	+		+															
11025			+	+																
16000				+	+	+	+													
22050						+	+	+	+											
32000							+		+	+	+									
44100							+		+		+	+	+	+	+	+	+	+	+	
48000												+		+	+	+	+			

除了这些预期的 WMA 解码配置文件外，还支持所有其他比特率和采样率组合，包括可变比特率 WMA 流。请注意，WMA 不像 MP3 那样均匀地消耗比特流，因此您需要更高的峰值传输能力才能以相同的比特率进行干净的播放。



8.2.3 支持的 RIFF WAV 格式

支持最常见的 RIFF WAV 子格式。

格式	姓名	支持的	评论
0x01	相变材料	+	16 和 8 位，任何样本 ra ≤ 特48千赫
0x02	ADPCM	-	
0x03	IEEE浮点型	-	
0x06	一条法律	-	
0x07	穆拉夫	-	
0x10	冲电气ADPCM	-	
0x11	IMA ADPCM	+	任意采样率 ≤48千赫
0x15	<small>数字地理信息技术发展部</small>	-	
0x16	数码修复	-	
0x30	杜比AC2	-	
0x31	GSM610	-	
0x3b	罗克韦尔ADPCM	-	
0x3c	罗克韦尔数码通讯	-	
0x40	G721.ADPCM	-	
0x41	G728CELP	-	
0x50	MPEG	-	
0x55	MPEG层3	+	有关支持的 MP3 模式，请参阅第 8 章。
0x64	G726.ADPCM	-	
0x65	G722.ADPCM	-	

2.1



8.2.4 支持的 MIDI 格式

播放通用 MIDI 和 SP-MIDI 格式 0 文件。用户必须将格式 1 和 2 文件转换为格式 0。最大同时复调数为 40。实际复调数取决于内部时钟速率（用户可选择）、使用的乐器以及可能启用的后处理效果，例如低音和高音增强器。复调限制算法利用 SP-MIDI MIP 表（如果存在）。

36.86兆赫（3.0×输入时钟）可实现 16-26 个同步持续音符。瞬时音符量可以更大。36 MHz 是功耗和质量之间的合理折衷，但可以使用更高的时钟来增强复调性。

VS1003b 实现了 36 种不同的仪器。每个旋律、效果和打击乐器都映射到其中一种乐器中。

VS1003b		
旋律	影响	打击乐器
钢琴	反钹	低音鼓
电颤琴	吉他音品噪音军鼓	
器官	呼吸	闭合踩镲
吉他	海滨	打开踩镲
失真吉他鸟鸣叫		高筒鼓
低音	电话	低音通鼓
小提琴	直升机	碎钹
字符串	掌声	骑钹
喇叭	枪声	手鼓
萨克斯管		高康加舞
长笛		低康加舞
带领		马拉卡斯
软垫		克拉夫斯
钢鼓		

2

8.3 VS1003数据流

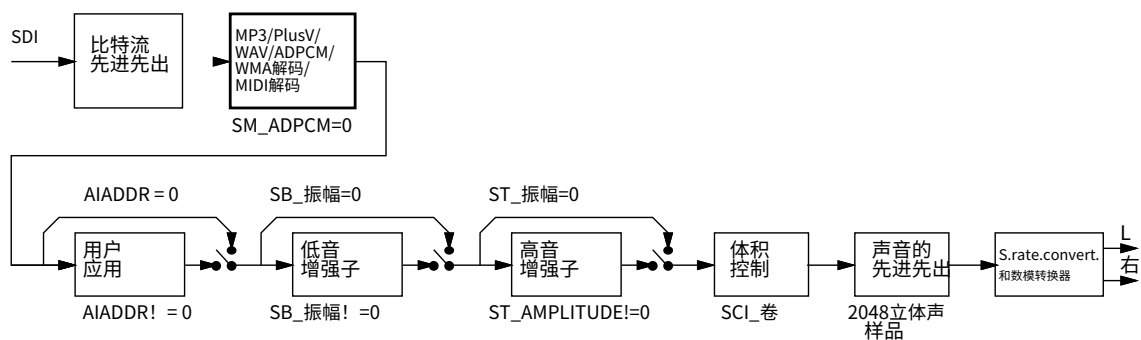


图 16: VS1003 的数据流。

首先，根据音频数据，如果未设置 ADPCM 编码模式，则从 SDI 总线接收并解码 MP3、WMA、PCM WAV、IMA ADPCM WAV 或 MIDI 数据。

解码后，如果 SCIAIADDR 非零，则从该寄存器指向的地址执行应用程序代码。有关更多详细信息，请参阅 VS10XX 的应用说明。

然后数据可能会根据 tBe 发送到低音和高音增强器 \_HA SSC 寄存器。

之后，信号被馈送到音量控制单元，该单元还将数据复制到音频 FIFO。

音频 FIFO 保存数据，这些数据由音频中断（第 10.13.1 章）读取并馈送到采样率转换器和 DAC。音频 FIFO 的大小为 2048 ste×re1o6-(b2it) 样本，或 8 KiB。

采样率转换器将所有不同的采样率转换为 XTALI/2，即最高可用采样率的 128 倍。这样就不再需要复杂的基于 PLL 的时钟方案，并且可以通过一个固定的输入时钟频率实现几乎无限的采样率精度。DA 转换器具有 12.288 MHz 时钟1t28×48kHz，即 6.144 MHz，并创建立体声同相模拟信号。过采样输出由片上模拟滤波器进行低通滤波。然后该信号被转发到耳机放大器。

8.4 串行数据接口（SDI）

串行数据接口用于传输压缩的 MP3 或 WMA 数据、WAV PCM 和 AD-PCM 数据以及 MIDI 数据。

如果 输入的 解码器无效或接收速度不够快，模拟输出自动米尤特d

还有 始终不同 ent 测试可以通过 SDI 激活，如第 9 章所述。

8.5 串行控制接口（SCI）

串行控制接口与SPI总线规范兼容。数据传输始终为 16 位。通过对接口寄存器的读写来控制VS1003。

控制界面主要控件有：

- 操作模式、时钟和内置效果的控制
- 访问状态信息和标头数据
- 访问编码的数字数据
- 上传用户程序

8.6 SCI 寄存器

SCI 寄存器，前缀 SCI					
注册	类型	重置	时间 <sup>1</sup>	缩写[位]	描述
0x0	读写	0x800	70 时钟 <sup>4</sup>	模式	模式控制
0x1	读写	0x3C <sup>3</sup>	40 时钟信号	地位	VS1003状态
0x2	读写	0	2100 时钟	低音	内置低音/高音增强器
0x3	读写	0	11000 西塔利 <sup>5</sup>	时钟	时钟频率+倍频
0x4	读写	0	40 时钟信号	解码时间 <sub>1</sub>	解码时间（秒）
0x5	读写	0	3200 时钟	奥达	杂项。音频数据
0x6	读写	0	80 时钟	随机存储器	内存写入/读取
0x7	读写	0	80 时钟	写入地址	RAM写入/读取的基地址
0x8	r	0	-	HDATA0	流头数据0
0x9	r	0	-	HDATA1	流头数据1
0xA	读写	0	3200 时钟 <sup>2</sup>	AIADDR	应用程序的起始地址
0xB	读写	0	2100 时钟	音量	音量控制
0xC	读写	0	50 时钟信号 <sup>2</sup>	人工智能控制0	应用控制寄存器0
0xD	读写	0	50 时钟信号 <sup>2</sup>	人工智能控制1	应用控制寄存器1
0xE	读写	0	50 时钟信号 <sup>2</sup>	人工智能控制2	应用控制寄存器2
0xF	读写	0	50 时钟信号 <sup>2</sup>	人工智能控制3	应用控制寄存器3

<sup>1</sup>这是写入该寄存器后 DREQ 保持低电平的最坏情况时间。对于执行时间少于 100 个时钟周期的寄存器写入，用户可以选择跳过 DREQ 检查。

<sup>2</sup>此外，还必须计算用户应用程序中所花费的周期。

<sup>3</sup>固件立即将该寄存器的值更改为 0x38，并在不到 100 毫秒的时间内更改为 0x30。

<sup>4</sup>当模式寄存器写入指定软件复位时，最坏情况时间为 16600 个 XTALI 周期。

<sup>5</sup>写入该寄存器可能会强制内部时钟运行1A。0t ×XTALI一段时间了。因此，在寄存器更新过程中发送 SCI 或 SDI 位并不是一个好主意。

请注意，如果 SCI 写入完成时 DREQ 为低电平，则 SCI 写入处理后 DREQ 也会保持低电平。



8.6.1 SCI 模式（RW）

SCI MODE 用于控制 VS1003 的操作，默认为 0x0800S(S 为 SM 寄存器集)。

少量	姓名	功能	价值	描述
0	单模差分	微分	0 1	正常同相奥迪 左声道反转
1	SM塞托泽罗	设置为零	0 1	正确的 错误的
2	SM重置	软复位	0 1	没有重置 重置
3	SM输出WAV	跳出WAV解码	0 1	不 是的
4	SM PDOWN	掉电	0 1	打开 掉电
5	SM测试	允许 SDI 测试	0 1	不允许 允许
6	短信流	流模式	0 1	不 是的
7	SM SETTOZERO2	设置为零	0 1	正确的 错误的
8	SM DACT	DCLK有效沿	0 1	上升 坠落
9	西迪奥德	SDI位顺序	0 1	高位优先 最后最高位
10	SMSD分享	共享SPI片选	0 1	不 是的
11	SM迪威	VS1002 原生 SPI 模式	0 1	不 是的
12	单模ADPCM	ADPCM 录音激活	0 1	不 是的
13	SM ADPCM HP	ADPCM 高通滤波器激活	0 1	不 是的
14	SM 线路输入	ADPCM 录音选择器	0 1	麦克风 线路输入

当设置 SMDIFF 时，播放器反转左声道输出。对于立体声输入，这会创建虚拟环绕声，而对于单声道输入，这会创建差分左/右信号。

通过设置SR来启动软件复位 \_MESET 为 1。该位自动清零。

如果您想在中间停止解码 WAV、WMA 或 MIDI 文件，请设置 SO \_MUTOFWAV，然后发送数据遵循 DREQ，直到 SMOUTOFWAV 被清除。SCH \_I DAT1 也将被清除。对于WMA对于 MIDI，继续发送流是最安全的，对于 WAV 发送零。

SM PDOWN 位将 VS1003 设置为软件掉电模式。请注意，软件断电的能效远不及通过 XRESET 引脚激活的硬件断电。

如果设置了 SM TESTS，则允许 SDI 测试。有关 SDI 测试的更多详细信息，请参阅第 9.8 章。



SM\_STREAM 激活VS1003 的流模式。在此模式下，应以尽可能均匀的间隔发送数据（数据块最好小于 512 字节），并且 VS1003 尽一切努力通过将播放速度更改为 5% 来保持其输入缓冲区半满。为了获得最佳音质，平均速度误差应在 0.5% 以内，比特率不应超过 160 kbit/s，并且不应使用 VBR。有关详细信息，请参阅 VS10XX 的应用说明。此模式不适用于 WMA 文件。

SM\_DACT 定义SDI 数据时钟的有效沿。当“0”时，在上升沿读取数据，当“1”时，在下降沿读取数据。

当 SMSDIORD 清零时，SDI 上的字节将作为默认的 MSb 首先发送。通过设置SS \_ DMORD，用户可以反转 SDI 的位顺序，即首先接收位 0，最后接收位 7。但是，字节仍按默认顺序发送。该寄存器位对 SCI 总线没有影响。

如果还设置了 SMSDINew，则设置 SMSDISHARE 会使 SCI 和 SDI 共享相同的片选，如第 7.2 章中所述。

设置 SMSDJNEW 将激活 VS1002 本机串行模式，如第 7.2.1 和 7.4.2 章中所述。请注意，当 VS1003 启动时，该位被设置为默认值。

通过同时激活 SMADPCM 和 SMRESET，用户将激活 IMA ADPCM 录音模式。更多信息请参阅 VS10XX 的应用说明。

如果 SM\_ADPCM\_HP 与 SAMDPCM 和 SMRESET 同时设置，ADPCM 模式将以高通滤波器启动。当存在大量背景噪音时，这可能有助于语音的清晰度。ADPCM 编码器频率响应产生的差异如图 17 所示。

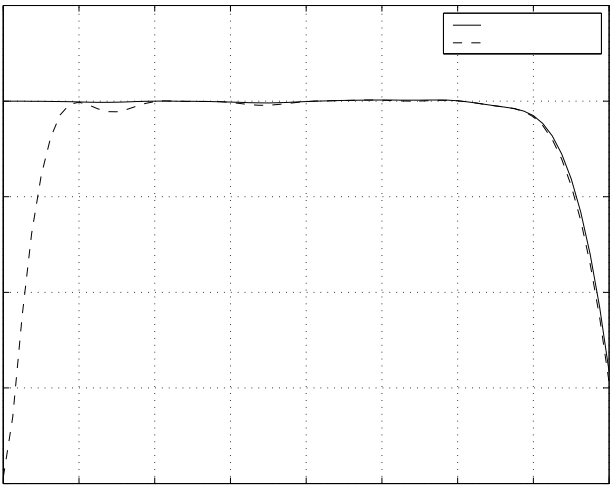


图 17: ADPCM 编码器频率响应 (8kHz 采样率)。

SM\_LINEIN 是使用 d 选择其他输入 f 哦 ADPCM 记录。如果为“0”，则 microphone 输入引脚 MICP 和 MICN are 用过的; 如果 '1', LINEIN 我使用。



8.6.2 SCI 状态 (RW)

SCI STATUS 包含有关 VS1003 当前状态的信息，让用户可以在没有音频故障的情况下关闭芯片。

姓名	位	描述
SSVER	6:4	版本
SSAPDOWN2	3	模拟驱动器断电
SSAPDOWN1	2	模拟内部掉电
萨沃	1:0	模拟音量控制

SSVER 对于 VS1001 为 0，对于 VS1011 为 1，对于 VS1002 为 2，对于 VS1003 为 3。

SSAPDOWN2 控制模拟驱动器断电。通常该位由系统固件控制。然而，如果用户希望以最小的断电瞬态对 VS1003 断电，请将此位设置为 1，然后在激活复位之前等待至少几毫秒。

SSAPDOWN1 控制内部模拟断电。该位仅供系统固件使用。

SSAVOL 是模拟音量控制：0 = -0 dB、1 = -6 dB、3 = -12 dB。该寄存器仅供系统固件自动使用。

8.6.3 SCI 低音 (RW)

姓名	位	描述
ST幅值	15:12	高音控制以 1.5 dB 为步长 (-8..7, 0 = 关闭)
ST频率限制	11:8	下限频率，以 1000 Hz 为步长 (0..15)
SB 幅度	7:4	以 1 dB 为步长的低音增强 (0..15, 0 = 离开)
SB频率限制	3:0	下限频率，以 10 Hz 为步长 (2..15)

低音增强器 VSBE 是一种强大的低音增强 DSP 算法，它试图在不造成削波的情况下充分利用用户耳机的功能。

当 SB 时 VSBE 被激活。振幅非零。SBAMPLITUDE 应设置为用户的偏好和 SF。BREQLIMIT 大约为用户音频系统最低频率的 1.5 倍。复制。例如设置 SB。CASS 到 0x00f6 将在 60 Hz 以下有 15 dB 增强。

注意：由于 VSBE 试图避免削波，因此它可以在动态音乐素材或播放音量未设置为最大时提供最佳低音增强。它也不会产生低音：源材料首先必须有一些低音。

当 SATMPLITUDE 非零时，高音控制 VSTC 被激活。例如设置 SC。超导系统至 0x7a00 将在 10 kHz 及以上具有 10.5 dB 高音增强。

低音增强器在 44100 Hz 采样率下使用约 3.0 MIPS 和高音控制 1.2 MIPS。两者可以同时打开。

8.6.4 SCI 时钟 (RW)

SCC的运作 VS1003 中的 I LOCKF 与 VS10x1 和 VS1002 中的不同。

SCI 时钟位		
姓名	位	描述
SC多路复用	15:13	时钟倍频器
SC添加	12:11	允许的乘数加法
SC频率	10: 0	时钟频率

SCMULT 激活内置时钟倍频器。这将乘以 XTALI 以创建更高的 CLKI。数值如下：

SC多路复用	面具	时钟信号
0	0x0000	西塔利
1	0x2000	西塔利 ×1。5
2	0x4000	西塔利 ×2。0
3	0x6000	西塔利 ×2。5
4	0x8000	西塔利 ×3。0
5	0xa000	西塔利 ×3。5
6	0xc000	西塔利 ×4。0
7	0xe000	西塔利 ×4。5

SCADD 告诉，解码器固件允许添加多少到指定的乘数 bM ySLCT  
如果暂时需要更多周期来解码 WMA 流。这些值为：

SC添加	面具	乘数加法
0	0x0000	不允许修改
1	0x0800	0.5 ×
2	0x1000	1.0 ×
3	0x1800	1.5 ×

SCFREQ 用于判断输入时钟 XTALI 是否以 12.288 MHz 以外的频率运行。XTALI 设置为 4 kHz 步长。计算该寄存器正确值的公式x  
特时间A是 11-8000000  
4000  
(XTALI 的单位为 Hz) 。

注意：假定默认值 0 表示 XTALI=12.288 MHz。

注意：因为最大采样率x我时间S阿里 —256,如果 XTAL 则所有采样率均不可用 <我12。288  
兆赫。

注意：自动时钟更改只能在解码 WMA 文件时发生。自动时钟更改已完成—0。5 ×一次。这不会导致掉落1。t0哦 ×时钟，您可以在整个 WMA 文件中使用相同的 SCI 和 SDI 时钟。当解码结束时，默认乘数将恢复，并可能导致1。0 ×暂时使用的时钟。

示例：如果 SCICLOCKF 为 0x9BE8，SCMULT = 4，SCADD = 3 且 SCFREQ = 0x3E8 = 1000。这意味着 XTALI =1000 ×4000+8000000 =12兆赫。时钟倍频器设置为t3哦。0 ×XTALI= 36MHz，固件可以自动选择使用的最大允许乘数为 (3。0+1。5) ×XTALI=54兆赫。



8.6.5 SCI 解码时间 (RW)

当解码正确数据时，当前解码时间以整秒的形式显示在该寄存器中。

用户可以更改该寄存器的值。在这种情况下，新值应写入两次。

SCI DECODETIME 在每次软件重置时以及 WAV (PCM 或 IMA ADPCM)、WMA 或 MIDI 解码开始或结束时重置。

8.6.6 SCI AUDATA (RW)

当解码正确数据时，当前采样率和通道数可以分别在 SCIAUDATA 的位 15:1 和 0 中找到。位 15:1 包含除以二的采样率，位 0 对于单声道数据为 0，对于立体声数据为 1。写信给 SA  
\_CUDATA 将直接改变采样率。

注意：由于错误，奇怪的采样率会反转 VS1003b 中立体声位的操作。

示例：44100 Hz 立体声数据读取为 0xAC45 (44101)。示例：  
11025 Hz 单声道数据读取为 0x2B10 (11025)。示例：11025  
Hz 立体声数据读取为 0x2B11 (11026)。  
示例：写入 0xAC80 将采样率设置为 44160 Hz，立体声模式不变。

8.6.7 SCI 写入存储器 (RW)

SCI WRAM 用于将应用程序和数据上传到指令和数据 RAM。起始地址必须通过写入 SC 来初始化  
\_第一次写入/读取 SCW 之前的 WIRAMADDR \_我内存。

由于 16 位数据可以用一个 SW 传输  
\_ CRAM 写/读，指令字为 32 位  
长，每个指令字需要两次连续的写入/读取。字节顺序是大尾数法（即最重要的字在前）。每次全字写入/读  
取后，内部指针会自动递增。

8.6.8 SCI 写入地址 (W)

SCI WRAMADDR 用于设置后续 SW 的程序地址  
\_ C 戳是写/读。地址  
偏移量 0 用于 X，0x4000 用于 Y，0x8000 用于指令存储器。也可以访问外设寄存器。

SM WRAMDDR 开始。。。结尾	目的地。地址。 开始。。。结尾	位/ 单词	描述
0x1800。。。0x187F	0x1800。。。0x187F	16	X 数据 RAM
0x5800。。。0x587F	0x1800。。。0x187F	16	Y 数据 RAM
0x8030。。。0x84FF	0x0030。。。0x04FF	32	指令存储器
0xC000。。。0xFFFF	0xC000。。。0xFFFF	16	输入/输出

上面仅列出了 X、Y 和指令存储器中的用户区域。其他区域可以访问，但不应写入，除非另有说明。



8.6.9 SCI HDAT0 和 SCI HDAT1 (R)

对于 WAV 文件，SCIHDAT0 和 SCIHDAT1 分别读取为 0x7761 和 0x7665。

对于 WMA 文件，SCIHDAT1 包含 0x574D，SCH I DAT0 包含以每秒字节数为单位的数据速度。要获得文件的比特率，请乘以 HSD 的值

– CAIT0 增 8。

对于 MIDI 文件，SCI HDAT1 包含 0x4D54，SCH I DAT0 包含根据下表的值：

HDAT0[15:8]	HDAT0[7:0]	价值	解释
0	复调音乐		当前复调音乐
1..255	预订的		

对于 MP3 文件，SCIHDAT[0。。。1]有以下内容：

少量	功能	价值	解释
HDAT1[15:5]	同步字	2047	流有效
HDAT1[4:3]	ID	3	ISO 11172-3 MPG 1.0
		2	ISO 13818-3 MPG 2.0 (1/2 级 e)
		1	MPG 2.5 (1/4 速率)
		0	MPG 2.5 (1/4 速率)
HDAT1[2:1]	层	3	我
		2	二
		1	三、
		0	预订的
HDAT1[0]	保护位	1	无CRC
		0	CRC 保护
HDAT0[15:12]	比特率		ISO 11172-3
HDAT0[11:10]	采样率	3	预订的
		2	32/16/8kHz
		1	48/24/12 kHz
		0	44/22/11 kHz
HDAT0[9]	填充位	1	附加插槽
		0	正常帧
HDAT0[8]	私人位		没有定义的
HDAT0[7:6]	模式	3	单核细胞增多症
		2	双通道
		1	联合立体声
		0	立体声
HDAT0[5:4]	扩大		ISO 11172-3
HDAT0[3]	版权	1	有版权的
		0	自由的
HDAT0[2]	原来的	1	原来的
		0	复制
HDAT0[1:0]	强调	3	CCITT J.17
		2	预订的
		1	50/15微秒
		0	没有任何

读取时，SCH I DAT0 和 SCIHDAT1 包含从 MP3 流中提取的标头信息



目前正在解码中。复位后，两个寄存器都被清除，表明尚未找到数据。

SCH 中的“采样率”字段 ID DAT0 的解释如下表：

“采样率”	ID=3/赫兹	ID=2/赫兹	ID=0,1/赫兹
3	-	-	-
2	32000	16000	8000
1	48000	24000	12000
0	44100	22050	11025

HDAT0中的“比特率”字段根据下表读取：

“比特率”	ID=3/kbit/s	ID=0,1,2/kbit/s
15	禁止的	禁止的
14	320	160
13	256	144
12	224	128
11	192	112
10	160	96
9	128	80
8	112	64
7	96	56
6	80	48
5	64	40
4	56	32
3	48	24
2	40	16
1	32	8
0	-	-

8.6.10 SCIAIADDR (RW)

SCI AIADDR表示之前用WSR编写的应用代码的起始地址和 SCIWRAM 寄存器。如果没有使用应用程序代码，则不应初始化该寄存器，或者应将其初始化为零。有关更多详细信息，请参阅 VS10XX 的应用说明。

蔡马德

### 8.6.11 SCIVOL (RW)

SCIVOL 是播放器硬件的音量控制。对于每个通道，可以定义 0..254 范围内的值来设置其相对于最大音量级别的衰减（以 0.5 dB 为步长）。然后将左通道值乘以 256，并将这些值相加。因此，最大音量为 0，总静音为 0xFEFE。

示例：左声道音量为 -2.0 dB，右声道音量为 -3.5 dB： $(4 \times 256) + 7 = 0x407$ 。请注意，启动时音量设置为全音量。重置软件不会重置音量设置。

注意：将 SCIVOL 设置为 0xFFFF 将激活模拟掉电模式。

### 8.6.12 SCIACTRL[x] (RW)

SCIACTRL[x] 寄存器 ( $x=[0..3]$ ) 可用于访问用户的应用程序。



## 9 手术

### 9.1 时钟

VS1003 在单个名义上 12.288 MHz 基频主时钟上运行。该时钟可由外部电路（连接到引脚 XTALI）或内部时钟晶体接口（引脚 XTALI 和 XTALO）生成。

### 9.2 硬件复位

当XRESET信号被驱动为低电平时，VS1003被复位并且所有控制寄存器和内部状态被设置为初始值。XRESET 信号与任何外部时钟异步。复位模式兼作完全掉电模式，其中 VS1003 的数字和模拟部分均处于最低功耗阶段，并且时钟停止。XTALO 也接地。

硬件复位（或上电时）后，DREQ 将保持至少 16600 个时钟周期，这意味着如果 VS1003 以 12.288 MHz 运行，则会有大约 1.35 ms 的延迟。之后用户需要设置SM等基本软件寄存器

解码。详细信息请参见第 8.6 节。

内部时钟可以通过 PLL 倍频。通过 C 支持的乘法器是 1.0 × 。 。 。 4.5 × 输入时钟。内部时钟倍频器的复位值 1 我。 s0 ×。如果需要典型值，则需要设置内部时钟乘法器 3t。 0 到 × 重置后。等待 DREQ 上升，然后将值 0x9800 写入 SCC  
\_I LOCKF（寄存器 3）。详细信息请参见第 8.6.4 节。

### 9.3 软件复位

在某些情况下，必须重置解码器软件。这是通过激活位 2Min 来完成的（第 8.6.1 章）。然后至少等待  $\mu$  t2，然后看 DREQ。DREQ 将保持低位至少 16600 时钟周期，这意味着如果 VS1003 以 12.288 MHz 运行，会有大约 1.35 ms 的延迟。DREQ 启动后，您可以照常继续播放。

如果您想确保 VS1003 不会切断低比特率数据流的结尾，并且您想要进行软件重置，建议在文件之后和重置之前向 SDI 总线提供 2048 个零（尊重 DREQ）。这对于 MIDI 文件尤其重要，尽管您也可以使用 HeDSAC  
轮询。

如果您想在中间中断 WAV、WMA 或 MIDI 文件的播放，请设置 SO 模式寄存器，并等待 SC 在继续之前，HIDAT1 被清除（有两秒超时）通过软件重置。MP3 目前未实现 OSU 格式，因此有超时要求。



## 9.4 ADPCM 录音

本章介绍如何创建 IMA ADPCM 格式的 RIFF/WAV 文件。这是一种广泛支持的 ADPCM 格式，许多 PC 音频播放程序都可以播放它。与线性 16 位音频相比，IMA ADPCM 录音的压缩比大致为 4:1。这使得以 32.44 kbit/s 的速度录制 8 kHz 音频成为可能。

### 9.4.1 激活 ADPCM 模式

IMA ADPCM 记录模式通过设置 SCI 模式中的 SRMESET 和 SMADPCM 位来激活。（可选）还可以通过设置 AgDSPM 启用高通滤波器以实现 8 kHz 采样率。CM HP 在同时。如果 SLM，则使用线路输入代替麦克风。INE IN 已设置。在激活 ADPCM 录音之前，用户**必须**将时钟分频器值写入 SCA 寄存器 ICTRL0 并获得 SCIAICTRL1。

使用 SM 的差异。ADPCM HP 如图 17（第 33 页）所示。作为一般规则，音频如果不使用 SMADPCM HP，将会更饱满、更接近原始。然而，当高通滤波器处于活动状态时，语音可能会更容易理解。仅使用采样率为 8 kHz 的滤波器。

在激活 ADPCM 记录之前，用户应将时钟分频器值写入 ASIC ICTRL0。这采样频率由以下公式计算：
$$F_s = \frac{F_c}{256 \times d}$$
 其中  $F_c$  是内部时钟（CLKI）和  $d$  是 SCA 中的分频值 ICTRL0。最低有效值  $d$  为 4。如果 SCIAICTRL0 包含 0，则使用默认分频器值 12。

例子：

$F_c = 2.0 \times 12$ . 288 兆赫兹， $d = 12$ . 现在  $F_s = \frac{2.0 \times 12}{256 \times 12} = 8000$  赫兹。  
 $F_c = 2.5 \times 14$ . 第 745 兆赫兹， $d = 18$ . 现在  $F_s = \frac{2.5 \times 14}{256 \times 18} = 8000$  赫兹。  
 $F_c = 2.5 \times 13$  兆赫兹， $d = 16$ . 现在  $F_s = \frac{2.5 \times 13}{256 \times 16} = 7935$  赫兹。

此外，在激活 ADPCM 模式之前，用户必须设置线性录音增益控制以寄存器 SCIAICTRL1。1024 等于数字增益 1，512 等于数字增益 0.5，依此类推。如果用户想要使用自动增益控制 (AGC)，SCIAICTRL1 应设置为 0。典型的语音应用通常最好使用 AGC，因为这可以保证录音中相对均匀的语音响度。

自 VS1033c SCA 寄存器 ICTRL2 控制最大 AGC 增益。如果 SCIAICTRL2 为零，则 maximum 增益为 65535 ( $6 \times 4$ )，即使用整个范围。这与之前的操作是兼容的。

### 9.4.2 读取 IMA ADPCM 数据

IMA ADPCM 录音激活后，寄存器 SHCDIAT0 和 SCIHDAT1 具有新功能。

IMA ADPCM 样本缓冲区有 1024 个 16 位字。缓冲区的填充状态可以从 SCIHDAT1 读取。如果 SCIHDAT1 大于 0，则可以从 HSDCAIT0 读取尽可能多的 16 位字。如果数据读取速度不够快，缓冲区就会溢出并返回到空状态。

注：如果 SCIHDAT1  $\geq 896$ 、在读取样本之前等待缓冲区溢出并清除可能会更好。这样您就可以避免缓冲区别名。

每个IMA ADPCM块是128个字，即256个字节。如果您希望中断读取数据并可能稍后继续，请在 128 字边界处停止。这样，整个块都会被跳过，编码流保持有效。

9.4.3 添加 RIFF 标头

要使 IMA ADPCM 文件成为 RIFF / WAV 文件，您必须在实际数据之前添加标头。请注意，2 字节和 4 字节值是小端字节序（最低字节在前），格式如下：

文件偏移量	字段名称	尺寸	字节	描述
0	块ID	4	“即兴”	
4	块大小	4	F0 F1 F2 F3	文件大小 - 8
8	格式	4	“海浪”	
12	子块1ID	4	“fmt”	
16	子块 1 大小	4	0x14 0x0 0x0 0x0	20
20	音频格式	2	0x11 0x0	0x11 用于 IMA ADPCM
22	频道数	2	0x1 0x0	单声道
24	采样率	4	R0 R1 R2 R3	8kHz 为 0x1f40
28	字节率	4	B0 B1 B2 B3	0xfd7 表示 8 kHz
32	块对齐	2	0x0 0x1	0x100
34	每个样本的位数	2	0x4 0x0	4位ADPCM
36	字节额外数据	2	0x2 0x0	2
38	额外数据	2	0xf9 0x1	每块样本 (505)
40	子块2ID	4	“事实”	
44	子块2大小	4	0x4 0x0 0x0 0x0	4
48	样本数	4	S0 S1 S2 S3	
52	子块3ID	4	“数据”	
56	子块3大小	4	D0 D1 D2 D3	数据大小（文件大小-60）
60	区块1	256		第一个 ADPCM 块
316	。 。 。			更多 ADPCM 数据块

如果我们有  $n$  音频块，表中数值如下： $F=n\times256+52$

$$\begin{aligned} F &= F_s \text{ (请参阅第 9.4.1 章了解如何计算 } F_s \text{ (拉.st)e)} \\ F &= \frac{F_s \times 256}{505} \\ S &= n \times 505, D = n \times 256 \end{aligned}$$

如果您事先知道要记录多少，则可以在任何实际数据之前填写完整的标题。但是，如果您不知道要记录多少，则必须填写标题大小数据  $F_s$ ,  $S$  和  $D$  录制完成后。

ADPCM 块的 128 个字（256 个字节）从  $S$  读取并写入文件。接下来，SCH 的高 8 位应该作为第一个字节写入文件，然后是低 8 位。请注意，这与某些 16 位微控制器的默认操作相反，您可能需要格外小心才能正确执行此操作。

查看是否以正确方式写入文件的一种方法是检查每个 256 字节块的字节 2 和 3（第一个字节计为字节 0）。字节 3 应始终为零。

#### 9.4.4 播放ADPCM数据

为了播放 IMA ADPCM 录音，您必须有一个带有第 9.4.3 章中所述标题的文件。如果是这种情况，您所需要的就是通过 SDI 提供 ADPCM 文件，就像处理任何音频文件一样。

#### 9.4.5 采样率注意事项

支持 IMA ADPCM 播放的 VS10xx 芯片能够以任何采样率播放 ADPCM 文件。然而，其他一些程序可能期望 IMA ADPCM 文件具有一些精确的采样率，例如 8000 或 11025 Hz。此外，某些程序或系统不支持低于 8000 Hz 的采样率。

但是，如果您没有合适的时钟，则可能无法获得精确的 8 kHz 采样率。如果您有 12 MHz 时钟，您可以获得的最接近的采样率  $2w$ 。0伊思  $\times 12$  兆赫兹和  $d=12$  是  $F_s=7812.5$  赫兹。由于频率误差仅为 2.4%，因此最好  $F_s=$  哦  $s=$  等 8000 赫兹到  
如果还要用 PC 播放同一文件，请添加标题。这会导致样本播放速度稍快一些（59 秒播放一分钟）。

但请注意，除非绝对必要，否则不应按照此处描述的方式调整采样率。

如果您希望以提高数据速率为代价获得更好的质量，则可以使用更高的采样率，例如 16 kHz。

#### 9.4.6 示例代码

以下代码初始化 VS1003b/VS1023 上的 IMA ADPCM 编码并显示如何读取数据。

```
const unsigned char header[] = {
    0x52, 0x49, 0x46, 0x46, 0x1c, 0x10, 0x00, 0x00,
    0x57, 0x41, 0x56, 0x45, 0x66, 0x6d, 0x74, 0x20, /*|RIFF....WAVEfmt|*/ 0x14,
    0x00, 0x00, 0x00, 0x11, 0x00, 0x01, 0x00,
    0x40, 0x1f, 0x00, 0x00, 0x75, 0x12, 0x00, 0x00, /*|.....@.....|*/ 0x00,
    0x01, 0x04, 0x00, 0x02, 0x00, 0xf9, 0x01,
    0x66, 0x61, 0x63, 0x74, 0x04, 0x00, 0x00, 0x00, /*|.....事实....|*/ 0x5c,
    0x1f, 0x00, 0x00, 0x64, 0x61, 0x74, 0x61,
    0xe8, 0x0f, 0x00, 0x00
};
```

无符号字符数据库[512]; /\* 用于保存到磁盘的数据缓冲区 \*/



```
void RecordAdpcm1003(void) { /* VS1003b/VS1033c */
    u_int16 w = 0, idx = 0;

    。 。 。 /* 检查并定位磁盘上的可用空间 */

    设置Mp3Vol(0x1414); /* 录音监听音量 */ 0); /* 禁用低音/
    WriteMp3SpiReg(SCI_BASS, 高音 */

    WriteMp3SpiReg(SCI_CLOCKF, 0x4430); /* 2.0x 12.288MHz */ 等待
    (100);
    WriteMp3SpiReg(SCI_AICTRL0, 12); 等待 /* 格 -> 12=8kHz 8=12kHz 6=16kHz */
    (100) ;
    WriteMp3SpiReg(SCI_AICTRL1, 0); 等待 /* 自动增益 */
    (100) ;
    如果 (行输入) {
        WriteMp3SpiReg(SCI_MODE, 0x5804); /* 正常软件复位 + 其他位 */ } else {

        WriteMp3SpiReg(SCI_MODE, 0x1804); /* 正常SW复位+其他位*/
    }
    for (idx=0; idx < sizeof(header); idx++) { /* 先保存 header */
        db[idx] = 标题[idx];
    }
    /* 如果需要的话修复速率
    */ /* db[24] = 速度; */
    /* 数据库[25] = 率 >> 8; */

    /* 录音循环 */ while
    (recording_on) {
        做 {
            w = ReadMp3SpiReg(SCI_HDAT1);
            while (w < 256 || w >= 896); /* 等待 512 字节可用 */

            而 (idx<512) {
                w = ReadMp3SpiReg(SCI_HDAT0);
                db[idx++] = w>>8;
                db[idx++] = w&0xFF;
            }
            idx = 0;
            write_block(datasector++, db); /* 将输出块写入磁盘 */
        }
        。 。 。 /* 修复WAV头信息 */ /* 然后更新FAT信
        。 。 。 息 */
        重置MP3(); /* 正常复位, 恢复默认设置 */
        SetMp3Vol(vol);
    }
}
```



9.5 SPI 启动

如果在启动时将 GPIO0 设置为上拉电阻为 1，则 VS1003 会尝试从外部 SPI 存储器启动。

SPI启动重新定义了以下引脚：

正常模式	SPI 启动模式
GPIO0	坐标系
GPIO1	时钟脉冲
DREQ	莫西
GPIO2	味噌

存储器必须是具有 16 位地址（即至少 1 KiB）的 SPI 总线串行 EEPROM。VS1003 使用的串行速度为 245 kHz，标称时钟为 12.288 MHz。内存中的前三个字节必须是 0x50、0x26、0x48。确切的记录格式在 VS10XX 的应用笔记中进行了解释。

9.6 播放/解码

这是VS1003的正常工作模式。SDI 数据被解码。解码样本由内部 DAC 转换为模拟域。如果没有找到可解码数据，则设置 HSDCAIT0 和 SCIH DAT1 为 0 且模拟输出静音。

当没有解码输入时，VS1003 进入空闲模式（比解码期间功耗更低）并主动监视串行数据输入中的有效数据。

所有不同的格式都可以连续播放，中间无需软件重置。每个流后至少发送 4 个零。然而，在流之间使用软件重置可能仍然是一个好主意，因为它可以防止损坏的文件。在这种情况下，您不应该等待解码完成H(D SACTI0 和

在发出软件复位之前，SCI HDAT1 变为零。

9.7 馈送 PCM 数据

通过向 VS1003 发送 WAV 文件头，可以将 VS1003 用作 PCM 解码器。如果 WAV 文件中发送的长度为 0 或 0xFFFFFFFF，VS1003 将无限期地保持在 PCM 模式（或直到 SOM UTOFWAV 已放）。单声道或立体声支持 8 位线性和 16 位线性音频。



9.8 SDI 测试

VS1003中有多种测试模式，允许用户执行内存测试、SCI总线测试和几种不同的正弦波测试。

所有测试都以类似的方式开始：VS1003是硬件复位，TS \_ EMTS设置完毕，然后进行测试命令被发送到SDI总线。每个测试都是通过发送一个 4 字节特殊命令序列开始的，后跟 4 个零。序列描述如下。

9.8.1 正弦测试

正弦测试使用 8 字节序列 0x53 0xEF 0 进行初始化n60E0 0 0，其中n定义要使用的正弦测试。n定义如下：

n位		
姓名	位	描述
Fsidx	7:5	采样率指数
S	4:0	正弦跳跃速度

Fsidx	Fs
0	44100赫兹
1	48000赫兹
2	32000赫兹
3	22050赫兹
4	24000赫兹
5	16000赫兹
6	11025赫兹
7	12000赫兹

现在可以计算要输出的正弦频率F来回=米Fs×s128。

示例：使用值 126（即 0b01111110）激活正弦测试。布雷亚nktiongits 组件，Fsidx=0乙011 = 3因此Fs=22050赫兹。S=0乙11110 = 30,因此最终的正弦频率F=22050赫兹×30128≈5168赫兹。

要退出正弦测试，请发送序列 0x45 0x78 0x69 0x74 0 0 0 0。

注：正弦测试信号经过数字音量控制，因此可以单独测试通道。

9.8.2 引脚测试

引脚测试通过 8 字节序列 0x50 0xED 0x6E 0x54 0 0 0 0 激活。此测试仅用于芯片生产测试。



9.8.3 内存测试

内存测试模式使用 8 字节序列 0x4D 0xEA 0x6D 0x54 0 0 0 0 进行初始化。在此序列之后，等待 500000 个时钟周期。结果可以从 SCI 寄存器中读取。在 SCI 寄存器中，位 15 表示测试完成，位 14:7 表示没用过，位 6 表示 Mux 测试成功者，位 5 表示好我内存，位 4 表示好 Y 内存，位 3 表示好的 X 内存，位 2 表示好我的 ROM，位 1 表示好 Y ROM，位 0 表示好的 X ROM。如果所有位都为 1，则表示一切都好。

位	面具	意义
15	0x8000	测试完成
14:7		没用过
6	0x0040	Mux测试成功者
5	0x0020	好我内存
4	0x0010	好Y内存
3	0x0008	好的X内存
2	0x0004	好我的ROM
1	0x0002	好Y ROM
0	0x0001	好的X ROM
	0x807f	一切都好

内存测试会覆盖 RAM 内存的当前内容。

9.8.4 SCI 测试

Sci 测试使用 8 字节序列 0x53 0x70 0x 进行初始化，其中 0x 表示要测试的寄存器号。读取给定寄存器的内容并将其复制到 SCI 寄存器。在 SCI 寄存器中，位 15 表示测试完成，位 14:7 表示没用过，位 6 表示 Mux 测试成功者，位 5 表示好我内存，位 4 表示好 Y 内存，位 3 表示好的 X 内存，位 2 表示好我的 ROM，位 1 表示好 Y ROM，位 0 表示好的 X ROM。如果所有位都为 1，则表示一切都好。

示例：如果  $n$  为 48，SCI 寄存器 0 的内容（SCI\_MIODE）被复制到 SCIHDAT0。

## 10 VS1003寄存器

### 10.1 谁需要阅读本章

当用户希望向 VS1003 添加一些自己的功能（例如 DSP 效果）时，需要用户软件。

然而，VS1003的大多数用户不需要担心编写自己的代码，也不需要担心本章，包括那些只从VLSI Solution 网站下载软件插件的用户。

### 10.2 处理器核心

VS DSP 是一个 16/32 位 DSP 处理器内核，还具有广泛的通用处理器功能。VLSI Solution 的免费 VSKIT 软件包包含为 thDeSVPS 处理器核心编写、模拟和调试汇编语言或扩展 ANSI C 程序所需的所有工具和文档。

VLSI 解决方案还提供完整的集成开发环境 VSIDE，以实现完整的调试功能。

### 10.3 VS1003 内存映射

VS1003 的内存映射如图 18 所示。

### 10.4 SCI 寄存器

第 8.6 章中描述的 SCI 寄存器可以在 0xC000..0xC00F 之间找到。除了这些寄存器外，地址0xC010还有一个，称为C \_ SDIANGE。

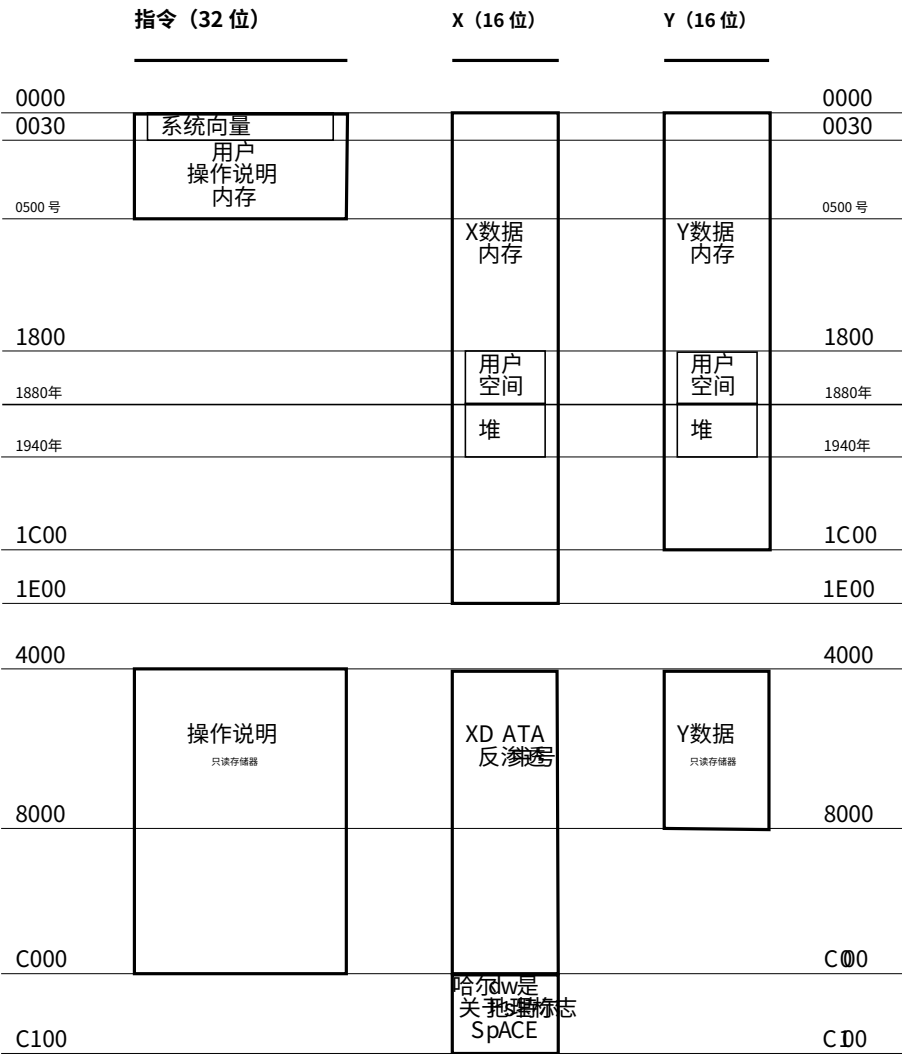
SCI 寄存器，前缀 SCI				
注册	类型	重置	缩写[位]	描述
0xC010	r	0	改变[5:0]	最后的 SCI 访问地址。

SCI 更改位		
姓名	位	描述
科学写 _	4	如果上次访问是写周期，则为 1。
SCI 通道地址	3:0	最后访问的 SPI 地址。

### 10.5 串行数据寄存器

SDI寄存器，前缀SER				
注册	类型	重置	缩写[位]	描述
0xC011	r	0	数据	最后收到 2 个字节，大端字节序。
0xC012	w	0	DREQ[0]	DREQ 引脚控制。





数字 18：用户的 中是里地图。

10.6 DAC寄存器

数模转换器 寄存器、pr e修复DAC _				
注册	类型	重置	缩写[位]	描述
0xC013	读写	0	FCTL	DAC 频率控制，16 LSb。
0xC014	读写	0	肝细胞癌	DAC频率系数 n控制4MSbs， 磷L 控制。
0xC015	读写	0	左边	DAC左声道P CM值。
0xC016	读写	0	正确的	DAC右声道 磷CM值。

每四个时钟周期，DAC 一个内部的 26 位计数器添加到 乙 (DEACTLH & 15) ×65536+ FCTL。每当 DAC 中断产生时 是反o verflows，来自 DA 的值在 DACRL GHT 被读取并 特德。



10.7 GPIO 寄存器

GPIO 寄存器，前缀 GPIO				
注册	类型	重置	缩写[位]	描述
0xC017	读写	0	DDR[3:0]	方向。
0xC018	r	0	数据[3:0]	从引脚读取的值。
0xC019	读写	0	奥数据[3:0]	设置到引脚的值。

GPIO DIR 用于设置GPIO 引脚的方向。1表示输出。GOPDIO 即使 GPIODIR 位设置为输入也是如此。ATA 记得它的

GPIO 寄存器不生成中断。

请注意，在 VS1003 中，可以通过 thW 读写 VSDSP 寄存器 SCI WRAM 寄存器。因此，您可以非常方便地使用 GPIO 引脚。eS 厘米地址和



10.8 中断寄存器

中断寄存器，前缀 INT				
注册	类型	重置	缩写[位]	描述
0xC01A	读写	0	启用[7:0]	中断使能。
0xC01B	w	0	全局 DIS[-]	写入以添加到中断计数器。
0xC01C	w	0	环球ENA[-]	写入以从中断计数器中减去。
0xC01D	读写	0	计数器[4:0]	中断计数器。

INT ENABLE 控制中断。控制位如下：

INT 启用位		
姓名	位	描述
INT EN TIM1	7	启用定时器 1 中断。
INT EN TIM0	6	启用定时器 0 中断。
INT EN 接收	5	使能 UART RX 中断。
INT EN TX	4	使能 UART TX 中断。
内部模块	3	使能 AD 调制器中断。
INT EN SDI	2	启用数据中断。
INT EN SCI	1	使能 SCI 中断。
INT EN DAC	0	使能 DAC 中断。

注意：更改 IE 可能需要最多 6 个时钟周期。神经网络没有任何影响。

将任何值写入 INTGLOB DIS 都会向中断计数器 INTCOUNTER 添加 1，并有效禁用所有中断。写入该寄存器可能需要最多 6 个时钟周期才会生效。

将任何值写入 INTGLOB ENA 都会使中断计数器减一（除非 IC 不调谐器已经是0）。如果中断计数器变为零，则选择中断埃文TBLE 是重新存储。中断例程应始终将写入该寄存器作为其执行的最后一件事，因为中断会自动向中断计数器加一，但将其减回其初始值是用户的责任。写入该寄存器可能需要最多 6 个时钟周期才能生效。

通过读取INTCOUNTER，用户可以检查中断计数器是否正确。如果寄存器不为 0，则禁用中断。



10.9 A/D 调制器寄存器

中断寄存器，前缀 AD				
注册	类型	重置	缩写[位]	描述
0xC01E	读写	0	DIV	A/D 调制器分频器。
0xC01F	读写	0	数据	A/D 调制器数据。

AD DIV 控制AD 转换器的采样频率。收集一个萨姆 $\frac{12p8乐}{n \times}$   $n$ 时钟周期被使用 $n$ （是 ADDIV 的值）。最低可用值为 4，当 CLKI 为 24.576 MHz 时，采样率为 48 kHz。当ADDIV 为0时，A/D转换器关闭。

AD DATA 包含最新解码的 A/D 值。



10.10 看门狗v1.0 2002-08-26

看门狗由看门狗计数器和一些逻辑组成。复位后，看门狗处于非活动状态。计数器重载值可以通过写入 WDO 来设置。看门狗由写入激活寄存器 0x4ea9 注册 WDOG 重置。每次执行此操作时，看门狗计数器都会重置。每一个第 65536 个时钟周期计数器减 1。如果计数器下溢，它将激活 vsdsp 的内部复位序列。

因此，在第一个 0x4ea9 写入 WDOR 之后，随后写入同一寄存器相同的值必须不小于  $e6v5e5 \times 36 \times \text{WDOG\_CONFIG}$  时钟周期。

一旦启动，看门狗就无法关闭。另外，写入 WDOG\_CONFIG 不会改变计数器重载值。

看门狗激活后，任何来自/向 WD 的读/写操作将使下一次对 WDOG 的写操作无效。这将防止失控循环重新设置计数器，即使他们碰巧写了正确的数字。向 WDOG\_RESET 写入错误值也会使下一次写入 WDOR 无效。格赛特。

从看门狗寄存器读取返回未定义的值。

10.10.1 寄存器

看门狗，前缀 WDOG				
注册	类型	重置	缩写	描述
0xC020	w	0	配置	配置
0xC021	w	0	重置	时钟配置
0xC022	w	0	假的[-]	虚拟寄存器



10.11 串口v1.0 2002-04-23

RS232 UART 使用 RS232 标准实现串行接口。

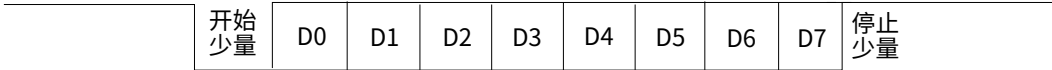


图 19：RS232 串行接口协议

当线路空闲时，它保持逻辑高状态。传输字节时，传输以起始位（逻辑零）开始，以数据位（LSB 在前）继续，以停止位（逻辑高）结束。每个 8 位字节帧发送 10 位。

10.11.1 寄存器

UART 寄存器，前缀 UARTx				
注册	类型	重置	缩写	描述
0xC028	r	0	状态[3:0]	地位
0xC029	读/写	0	数据[7:0]	数据
0xC02A	读/写	0	数据[15:8]	数据高
0xC02B	读/写	0	DIV	分频器

10.11.2 状态 UARTxSTATUS

从状态寄存器读取数据会返回发送器和接收器的状态。

UARTx 状态位		
姓名	位	描述
UART ST RXORUN	3	接收器溢出
UART ST RXFULL	2	接收器数据寄存器已满
UART ST TXFULL	1	发送器数据寄存器已满
UART ST TX 运行	0	发射机运行

如果接收的字节在从接收器移位寄存器传输到数据寄存器时覆盖未读的数据，则 UART ST RXORUN 被置位，否则被清零。

如果数据寄存器中有未读数据，则 UART ST RXFULL 被置位。

如果不允许写入数据寄存器（数据寄存器已满），则 UART ST TXFULL 被置位。

如果发送器移位寄存器正在运行，则 UART ST TXRUNNING 被置位。



10.11.3 数据 UARTxDATA

从 UARTxDATA 读取将返回位 7:0 中接收到的字节，位 15:8 返回为“0”。如果没有更多的数据要读取，接收器数据寄存器已满指示器将被清除。

当一个字节从接收器移位寄存器移动到接收器数据寄存器时，将产生接收中断。

对 UARTx DATA 的写入设置一个用于传输的字节。数据取自位 7:0，写入值中的其他位将被忽略。如果发送器空闲，则该字节立即移至发送器移位寄存器，生成发送中断请求，并开始发送。如果发送器忙，则 UART ST TXFULL 将被置位，该字节将保留在发送器数据寄存器中，直到发送完前一个字节并且可以继续发送为止。

10.11.4 数据高电平 UARTxDATAH

与 UARTxDATA 相同，但使用位 15:8。

10.11.5 分频器 UARTx DIV

UARTx DIV 位		
姓名	位	描述
UART 分区 D1	15:8	分频器 1 (0..255)
UART 分区 D2	7:0	分频器 2 (6..255)

复位时分频器设置为 0x0000。ROM 引导代码必须根据主时钟频率正确初始化它以获得正确的位速度。第二次迪D视频他必须为 6 到 255。

通讯速度 $Fd = (D1+1) \times (D2)$  ,在哪里 $F*$ 是主时钟频率， $Fdis$  TX/RX 速度（以 bps 为单位）。

26 MHz 主时钟下常见通信速度的分频器值：

UART 速度示例， $F*=26$ 兆赫兹		
通讯。速度 [bps]	UART DIV D1	UART 分区 D2
4800	85	63
9600	42	63
14400	42	42
19200	51	26
28800	42	21
38400	25	26
57600	1	226
115200	0	226

### 10.11.6 中断和操作

发送器的操作如下：将一个 8 位字写入发送数据寄存器后，如果发送器不忙于发送前一个字节，则该字将立即发送。当传输开始时，将发送 TXINTR 中断。状态位 [1] 通知发送器数据寄存器为空（或满状态），位 [0] 通知发送器（移位寄存器）为空状态。如果发送器数据寄存器不为空（位 [1] = '0'），则不得将新字写入发送器数据寄存器。一旦移位到发送器并开始发送，发送器数据寄存器将被清空。每次产生发送中断时，将新字写入发送器数据寄存器是安全的。

接收器的操作如下：它对 RX 信号线进行采样，如果检测到从高到低的转换，则找到起始位。此后，它在位时间的中间对每个 8 位进行采样（使用恒定定时器），并首先填充接收器（移位寄存器）LSB。最后，如果检测到停止位（逻辑高），则接收器中的数据将移至接收数据寄存器，并发送 INRTXR 中断并设置状态位 [2]（接收数据寄存器已满），并且状态位 [2] 旧状态被复制到位 [3]（接收数据溢出）。之后接收器返回空闲状态等待新的起始位。读取接收器数据寄存器时，状态位 [2] 被清零。

RS232 通信速度使用两个时钟分频器设置。基本时钟是处理器主时钟。这些寄存器中的位 15-8 用于第一个分频器，位 7-0 用于第二个分频器。RX 采样频率是第二个分频器输入的时钟频率。



10.12 定时器v1.0 2002-04-23

有两个 32 位定时器，可以相互独立地初始化和使能。如果启用，定时器将初始化为其由处理器写入的起始值，并开始在每个时钟周期递减。当该值超过零时，将发送中断，并且定时器初始化为其起始值寄存器中的值，并继续向下计数。只要定时器被启用，它就会保持在该循环中。

定时器有一个用于向下计数的 32 位定时器寄存器和一个 32 位 TIMELR 寄存器用于保存定时器起始值由处理器写入。定时器还有一个 2 位 TIM 寄存器用于保存定时器使能。每个定时器是由使能寄存器的相应位启用（1）或禁用（0）。

10.12.1 寄存器

定时器寄存器，前缀TIMER				
注册	类型	重置	缩写	描述
0xC030	读/写	0	配置[7:0]	定时器配置
0xC031	读/写	0	启用[1:0]	定时器使能
0xC034	读/写	0	T0L	Timer0 起始值 - LSB
0xC035	读/写	0	T0H	定时器 0 起始值 - MSB
0xC036	读/写	0	T0CNTL	Timer0 计数器 - LSB
0xC037	读/写	0	T0CTH	Timer0 计数器 - MSB
0xC038	读/写	0	T1L	Timer1 起始值 - LSB
0xC039	读/写	0	T1H	定时器 1 起始值 - MSB
0xC03A	读/写	0	T1CNTL	Timer1 计数器 - LSB
0xC03B	读/写	0	T1CNTH	Timer1 计数器 - MSB

10.12.2 配置定时器配置

定时器配置位		
姓名	位	描述
定时器 CF CLKDIV	7:0	主时钟分频器

TIMER CF CLKDIV 是所有定时器时钟的主时钟分频器。生成的内部时钟频率  $F_{我} = F_{*} / C_{d+1}$ ，在哪里  $F_{*}$  是主时钟频率  $C_{d}$  是定时器 CF\_CLKDIV。例子：对于 12 MHz 主时钟，TIMERC DIV=3 将主时钟除以 4，输出/采样因此时钟将是  $F_{我} = 12 \text{兆赫兹} / 4 = 3 \text{兆赫兹}$ 。



10.12.3 配置定时器启用

定时器使能位		
姓名	位	描述
定时器 EN T1	1	启用定时器1
定时器 EN T0	0	启用定时器0

10.12.4 定时器 X 起始值 TIMER Tx[L/H]

定时器复位时，32 位起始值 TIMERTx[L/H] 设置初始计数器值。计时器中断频率  $F_{\text{中断}} = \frac{F_{\text{我}}}{C+1}$  在哪里  $F_{\text{我}}$  是通过时钟分频器获得的主时钟（参见章节 10.12.2 之三和  $C$  是定时器 Tx[L/H]。

示例：使用 12 MHz 主时钟和 TIMEC \_ RF.CLKDIV=3，主时钟  $F_{\text{我}}=3$  兆赫兹。如果 TIMER TH=0，TIMER TL=99，则定时器中断频率  $F_{\text{中断}} = \frac{F_{\text{我}}}{C+1} = \frac{3\text{兆赫兹}}{100} = 30$  千赫。

10.12.5 定时器 X 计数器 TIMER TxCNT[L/H]

TIMER TxCNT[L/H] 包含当前计数器值。通过读取该寄存器对，用户可以了解距离下一次定时器中断还需要多长时间。另外，通过写入该寄存器，可以实现一次性不同长度定时器中断延迟。

10.12.6 中断

每个定时器都有自己的中断，当定时器计数器下溢时该中断被置位。



## 10.13 系统矢量标签

系统向量标签是可由用户替换以控制多个解码器功能的标签。

### 10.13.1 音频整数，0x20

通常包含以下VSDSP汇编代码：

```
jmp i DAC_INT_ADDRESS,(i6)+1
```

用户可以随意将第一条指令替换为  
打断。

杰米/圆速率获得对音频的控制的命令

### 10.13.2 科学整数，0x21

通常包含以下VSDSP汇编代码：

```
jmp i SCI_INT_ADDRESS,(i6)+1
```

用户可以随意将指令替换为杰米

A/圆速率命令来获得对 SCI 中断的控制。

### 10.13.3 数据整数，0x22

通常包含以下VSDSP汇编代码：

```
jmp i SDI_INT_ADDRESS,(i6)+1
```

用户可以随意将指令替换为杰米

A/圆速率命令来获得对 SDI 中断的控制。

### 10.13.4 Modulnt, 0x23

通常包含以下VSDSP汇编代码：

```
jmp i MODU_INT_ADDRESS,(i6)+1
```

用户可以随意将指令替换为杰米  
转换器中断。

A/圆速率命令来获得对AD模块的控制



#### 10.13.5 发送端, 0x24

通常包含以下VSDSP汇编代码: \_

```
jmp EMPTY INT_ADDRESS,(i6)+1
```

用户可以随意将指令替换为杰米  
打断。

A圆/周率获得对 UART TX 的控制的命令

#### 10.13.6 接收整数, 0x25

通常包含以下VSDSP汇编代码: \_

```
jmp RX INT_ADDRESS,(i6)+1
```

用户可以随意将第一条指令替换为  
RX 中断。

杰米/圆/周率获得对 UART 控制的命令

#### 10.13.7 定时器0Int, 0x26

通常包含以下VSDSP汇编代码: \_

```
jmp EMPTY INT_ADDRESS,(i6)+1
```

用户可以随意将第一条指令替换为  
0 中断。

杰米/圆/周率命令来获得对定时器的控制

#### 10.13.8 Timer1Int, 0x27

通常包含以下VSDSP汇编代码: \_

```
jmp EMPTY INT_ADDRESS,(i6)+1
```

用户可以随意将第一条指令替换为  
1 中断。

杰米/圆/周率命令来获得对定时器的控制

### 10.13.9 用户编解码器, 0x0

通常包含以下VSDSP汇编代码：

少年  
不

如果用户想要从标准解码器中夺取控制权，则应将第一条指令替换为适当的指令，命令到用户自己的代码。

除非用户同时输入 MP3 或 WMA 数据，否则系统会在不到 1 ms 的时间内激活用户程序。此后，用户应该从系统中窃取中断向量，并插入用户程序。

## 10.14 系统向量函数

系统向量函数是指向用户可以调用以帮助实现自己的应用程序的一些函数的指针。

### 10.14.1 WriteIRam(), 0x2

VS DSP C原型：

```
void WriteIRam(寄存器 i0 u int16 *addr, 寄存器 a1 u int16 msW, 寄存器 a0 u int16 lsW);
```

这是写入用户指令 RAM 的首选方法。

### 10.14.2 ReadIRam(), 0x4

VS DSP C原型：

```
u.int32 ReadIRam(register i0 u int16 *addr);
```

这是从用户指令 RAM 中读取数据的首选方法。

A1 包含结果的 MSB，a0 包含结果的 LSB。

### 10.14.3 数据字节(), 0x6

VS DSP C原型：

uint16 DataBytes(void);

如果用户通过切换UserCodec中的指针指向自己的代码来接管系统的正常操作，则可以通过该函数和以下两个函数从数据接口读取数据。

该函数返回可以读取的数据字节数。

#### 10.14.4 获取数据字节 () , 0x8

VS DSP C原型:

uint16 GetDataByte(void);

从数据接口读取并返回一个数据字节。该函数将等待，直到输入缓冲区中有足够的数据。

#### 10.14.5 获取数据字 () , 0xa

VS DSP C原型:

无效 GetDataWords(寄存器 i0 \_\_yu int16 \*d, 寄存器a0 u int16 n);

读n数据字节对并以大端格式复制它们（第一个字节到MSBs。）此函数将等待，直到输入缓冲区中有足够的数据。

#### 10.14.6 重启(), 0xc

VS DSP C原型:

无效重新启动（无效）；

导致软件重新启动，即跳转到标准固件而不重新初始化 IRAM 矢量。

这与软件重置功能不同，软件重置功能会导致完全初始化。

## 11 文档版本变更

本章描述了本文档最重要的更改。

### 1.04版，2009-02-03

- 第 4.7 节中添加了典型特性，第 4.3 节中更改了一些值。

### 版本1.03，2008-07-21

- 最大 SCI 读取时钟从 CLKI/6 更改为 CLKI/7。
- 更新了典型连接图。
- 如果 DREQ 为低电平，SCI 命令需要固定延迟。
- AD DIV 文档已修复。

### 版本1.02，2006-07-13

- 对 ADPCM 录音的一些说明。
- GBUF 现在称为共模缓冲器。
- 更新了第 6 节中的连接图

### 版本1.01，2005-12-08

- 添加了 ADPCM 录音部分（第 9.4 节）
- 将输出电压电流更改为 1 mA，最大 CLKI 更改为 52 MHz，温度范围 -40...85

### 版本1.00，2005-09-05

- AVDD 最大值降低至 2.85 V
- 生产版本，不再是初步版本

### 版本0.93，2005-06-23

- 更新功耗限制

### 版本0.92，2005-06-07

- 许可条款已更新
- 列出 Midi 乐器
- 推荐温度范围 -25...+70

## 12 联系信息

VLSI解决方案公司  
二楼G入口  
赫米安卡图 8 号  
FIN-33720 坦佩雷  
芬兰

传真: +358-3-3140-8288  
电话: +358-3-3140-8200  
电子邮件: [sales@vlsi.fi](mailto:sales@vlsi.fi)  
网址: <http://www.vlsi.fi/>