VS1001G - MPEG AUDIO CODEC

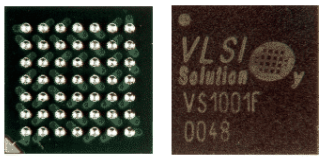
Features

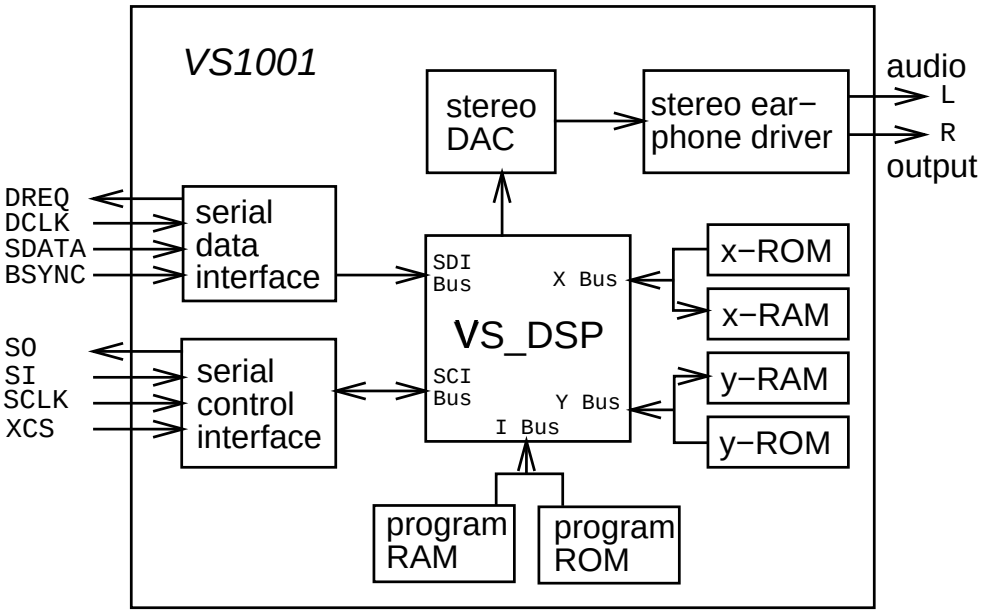
- MPEG audio layer 1, 2, and 3 decoder (ISO 11172-3)
- Supports MPEG 1 & 2 for all layers, and Layer 3's 2.5 extensions, and all their samplerates and bitrates, in mono or stereo
- Supports VBR (variable bit-rate)
- Operates with single clock at 12.288..16 MHz or 24.576..32 MHz (for lower bitrates also lower clocks may be used)
- Extremely low-power operation
- On-chip high-quality stereo DAC with no phase error between channels
- Internal Op-Amp in BGA-49 package
- Stereo earphone driver capable of driving a 30Ω load.
- 2.7 .. 3.6V operating voltage for analog
- 2.1 .. 3.6V operating voltage for digital
- 4 kB On-chip RAM for user code
- Serial control and data interfaces
- New functions (PCM input, streaming, etc) may be added with software

Description

VS1001G is a single-chip solution for an MPEG layer 3 audio decoder including a digital to analog converter and earphone amplifier. The chip contains a high-performance low-power DSP processor core (VS_DSP), working memory, 4 kBytes of program RAM and 0.3 kBytes of data RAM for user applications, serial control and input data interfaces, and a high-quality oversampling variable-samplerate stereo DAC.

The VS1001G receives its input bitstream through a serial input bus. The input stream is decoded and passed to an 18-bit oversampling, multi-bit, sigma-delta DAC. The decoding is controlled via a serial control bus. In addition to the basic decoding, it is possible to add application specific features, like DSP effects, to the program RAM memory of VS1001G.





Contents

List of Figures

1 Characteristics & Specifications

Unless otherwise noted: AVDD=3.4V, DVDD=2.7V, TA=+25°C, XTALI=24.576MHz, Full-Scale Output Sinewave at 1.125 kHz, measurement bandwidth 20..20000 Hz, bitstream 128 kbit/s, local components as shown in Figures 3 and 4.

1.1 Analog Characteristics

Parameter	Symbol	Min	Typ	Max	Unit
DAC Resolution			16		bits
DAC Differential Nonlinearity	DNL			±0.9	LSB
Total Harmonic Distortion	THD		0.1	0.2	%
Instantaneous Dynamic Range (DAC unmuted, A-weighted)	IDR	85	90		dB
Interchannel Isolation		60	75		dB
Interchannel Gain Mismatch				0.2	dB
Frequency Response		-0.1		0.1	dB
Full Scale Output Voltage (Peak-to-peak)			1.8[1]		Vpp
Gain Drift			100		ppm/°C
Deviation from Linear Phase				5	°
Out of Band Energy			-60		dB
Out of Band Energy with Analog Filter			-90		dB
Analog Output Load Resistance			30		Ω
Analog Output Load Capacitance				1000	pF
Power Supply Rejection			40		dB
Power Supply Consumption AVDD, Reset			0.1	1.0	μA
Power Supply Consumption AVDD, no load		3.0	4.5	6.0	mA
Power Supply Consumption AVDD, output loaded at 30Ω		4.0	5.5	40.0	mA
Power Supply Consumption AVDD, o. @ 30Ω + GND-buf.		6.0	7.5	40.0	mA
Power Supply Consumption DVDD, Reset			0.5	2.0	μA
Power Supply Consumption DVDD			15.0		mA

[1] 3.6 volts can be achieved with +-to-+ wiring for mono sound.

1.2 DAC Interpolation Filter Characteristics

Parameter	Symbol	Min	Typ	Max	Unit
Passband (to -3dB corner)		0		0.459Fs	Hz
Passband (Ripple Spec)		0		0.420Fs	Hz
Passband Ripple				±0.056	dB
Transition Band		0.420Fs		0.580Fs	Hz
Stop Band		0.580Fs			Hz
Stop Band Rejection		90			dB
Group Delay			15/Fs		s

Fs is conversion frequency

1.3 Absolute Maximum Ratings

Parameter	Symbol	Min	Max	Unit
Analog Positive Supply	AVDD	-0.3	3.6	V
Digital Positive Supply	DVDD	-0.3	3.6	V
Current at Any Digital Output			±50	mA
Voltage at Any Digital Input		DGND-1.0	DVCC+1.0	V
Ambient Operating Temperature (power applied)		-20	80	°C
Storage Temperature		-65	150	°C

1.4 Recommended Operating Conditions

Parameter	Symbol	Min	Typ	Max	Unit
Analog and Digital Ground	AGND DGND		0.0		V
Positive Analog (Functional)	AVDD	2.7	3.4	3.6	V
Positive Analog (In Spec)	AVDD	3.4	3.4	3.6	V
Ambient Operating Temperature		0		70	°C

The following values are to be used when the clock-doubler is active:

Parameter	Symbol	Min	Typ	Max	Unit
Positive Digital	DVDD	2.2	2.7	3.6	V
Input Clock Frequency[1]	CLKF		12.288	14	MHz

Parameter	Symbol	Min	Typ	Max	Unit
Positive Digital	DVDD	2.7	3.0	3.6	V
Input Clock Frequency[1]	CLKF		12.288	16	MHz

[2] The maximum samplerate that may be decoded with correct speed is CLKF/256.

The following values are to be used when the clock-doubler is inactive:

Parameter	Symbol	Min	Typ	Max	Unit
Positive Digital	DVDD	2.7	3.0	3.6	V
Input Clock Frequency[2]	CLKF		24.576	26	MHz

[2] The maximum samplerate that may be decoded with correct speed is CLKF/512.

1.5 Digital Characteristics

Parameter	Symbol	Min	Typ	Max	Unit
High-Level Input Voltage			2.25		V
Low-Level Input Voltage				0.8	V
High-Level Input Voltage at I_O = -2.0 mA		0.9DVDD			V
Low-Level Output Voltage at I_O = -2.0 mA				0.1DVDD	V
Input Leakage Current				1.0	μA

1.6 Switching Characteristics - Clocks

Parameter	Symbol	Min	Typ	Max	Unit
Master Clock Frequency [1]	XTALI		12.288		MHz
Master Clock Frequency [2]	XTALI		24.576		MHz
Master Clock Duty Cycle		40	50	60	%
Clock Output	XTALO		XTALI		MHz

[1] Clock doubler active.
[2] Clock doubler inactive.

1.7 Switching Characteristics - DREQ Signal

Parameter	Symbol	Min	Typ	Max	Unit
Data Request Signal	DREQ			200	ns

1.8 Switching Characteristics - SPI Interface Output

Parameter	Symbol	Min	Typ	Max	Unit
SPI Input Clock Frequency[1]				0.25×CLKF	MHz
Rise time for SO				100	ns

[1] If clock doubler is in use, the maximum speed is 0.5×CLKF.

1.9 Switching Characteristics - Boot Initialization

Parameter	Symbol	Min	Max	Unit
_RESET active time		2		clocks
_RESET inactive to software ready			2.0	ms

1.10 Short-Circuiting Analog Outputs

Although not a recommended practise for prolonged times, short-circuiting analog outputs and/or ground does not cause physical harm to the chip. Thus, there is not need to connect protective resistors for headphone connectors.

2 Pin Descriptions

2.1 SOIC-28

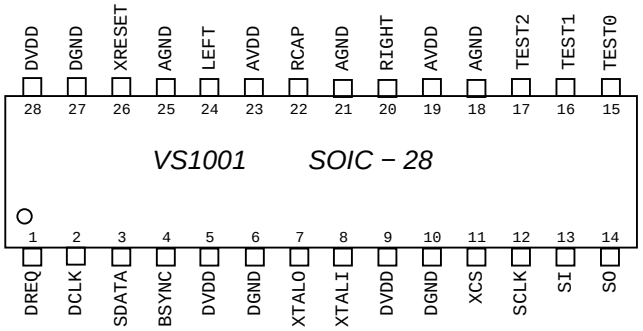


Figure 1: Pin Configuration, SOIC-28.

Pin Name	Pin	Type	Function
DREQ	1	DO	data request, input bus
DCLK	2	DIO	serial input data bus clock
SDATA	3	DI	serial data input
BSYNC	4	DI	byte synchronization signal
DVDD1	5	PWR	digital power supply
DGND1	6	PWR	digital ground
XTALO	7	CLK	crystal output
XTALI	8	CLK	crystal input
DVDD2	9	PWR	digital power supply
DGND2	10	PWR	digital ground
XCS	11	DI	chip select input (active low)
SCLK	12	DI	clock for serial bus
SI	13	DI	serial input
SO	14	DO3	serial output
TEST0	15	DI	reserved for test, connect to DVDD
TEST1	16	DIO	reserved for test, do NOT connect!
TEST2	17	DIO	reserved for test, do NOT connect!
AGND1	18	PWR	analog ground
AVDD1	19	PWR	analog power supply
RIGHT	20	AO	right channel output
AGND2	21	PWR	analog ground
RCAP	22	AIO	capacitance for reference
AVDD2	23	PWR	analog power supply
LEFT	24	AO	left channel output
AGND3	25	PWR	analog ground
XRESET	26	DI	active low asynchronous reset
DGND3	27	PWR	digital ground
DVDD3	28	PWR	digital power supply

Pin types:

- DI = digital input, CMOS Input Pad
- DO = digital output, CMOS Input Pad
- DIO = digital input/output
- DO3 = digital output, CMOS Tri-stated Output Pad
- AO = analog output
- CLK = clock/chrystal connection
- PWR = power supply pin

2.2 BGA-49

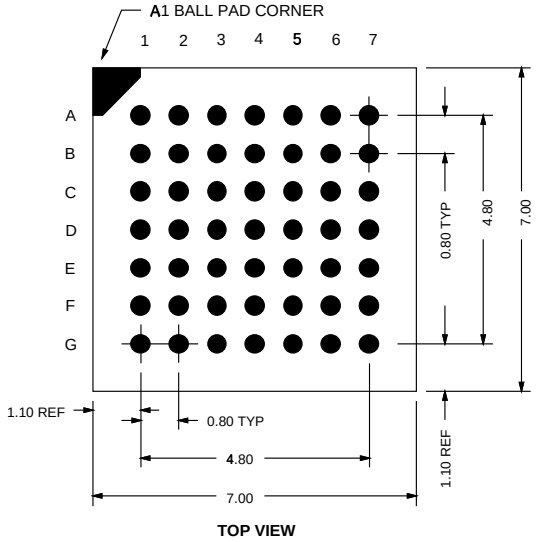


Figure 2: Pin Configuration, BGA-49.

Pin Name	Ball	Type	Function
BSYNC	E3	DI	byte synchronization signal
DVDD1	F3	PWR	digital power supply
DGND1	F4	PWR	digital ground
XTAL0	G3	CLK	crystal output
XTALI	E4	CLK	crystal input
DVDD2	F5	PWR	digital power supply
DGND2	F6	PWR	digital ground
XCS	G6	DI	chip select input (active low)
SCLK	D6	DI	clock for serial bus
SI	E7	DI	serial input
SO	D5	DO3	serial output
TEST0	C6	DI	reserved for test, connect to DVDD
TEST1	C7	DIO	reserved for test, do NOT connect!
TEST2	B6	DIO	reserved for test, do NOT connect!
AGND1	C5	PWR	analog ground
AVDD1	B5	PWR	analog power supply
RIGHT	A6	AO	right channel output
AGND34	B4	PWR	analog ground
GBGND	A5	PWR	analog ground for ground buffer
GBUF	C4	PWR	ground buffer
GBVDD	A4	PWR	analog power supply for ground buffer
RCAP	B3	AIO	capacitance for reference
AVDD45	A3	PWR	analog power supply
LEFT	B2	AO	left channel output
AGND56	A2	PWR	analog ground
XRESET	B1	DI	active low asynchronous reset
DGND3	D2	PWR	digital ground
DVDD3	D3	PWR	digital power supply
DREQ	E2	DO	data request, input bus
DCLK	E1	DIO	serial input data bus clock
SDATA	F2	DI	serial data input

For pin types, look at Chapter 2.1.

3 Connection Diagram, SOIC-28

In this connection diagram, a SOIC-28 -packaged VS1001G is used.

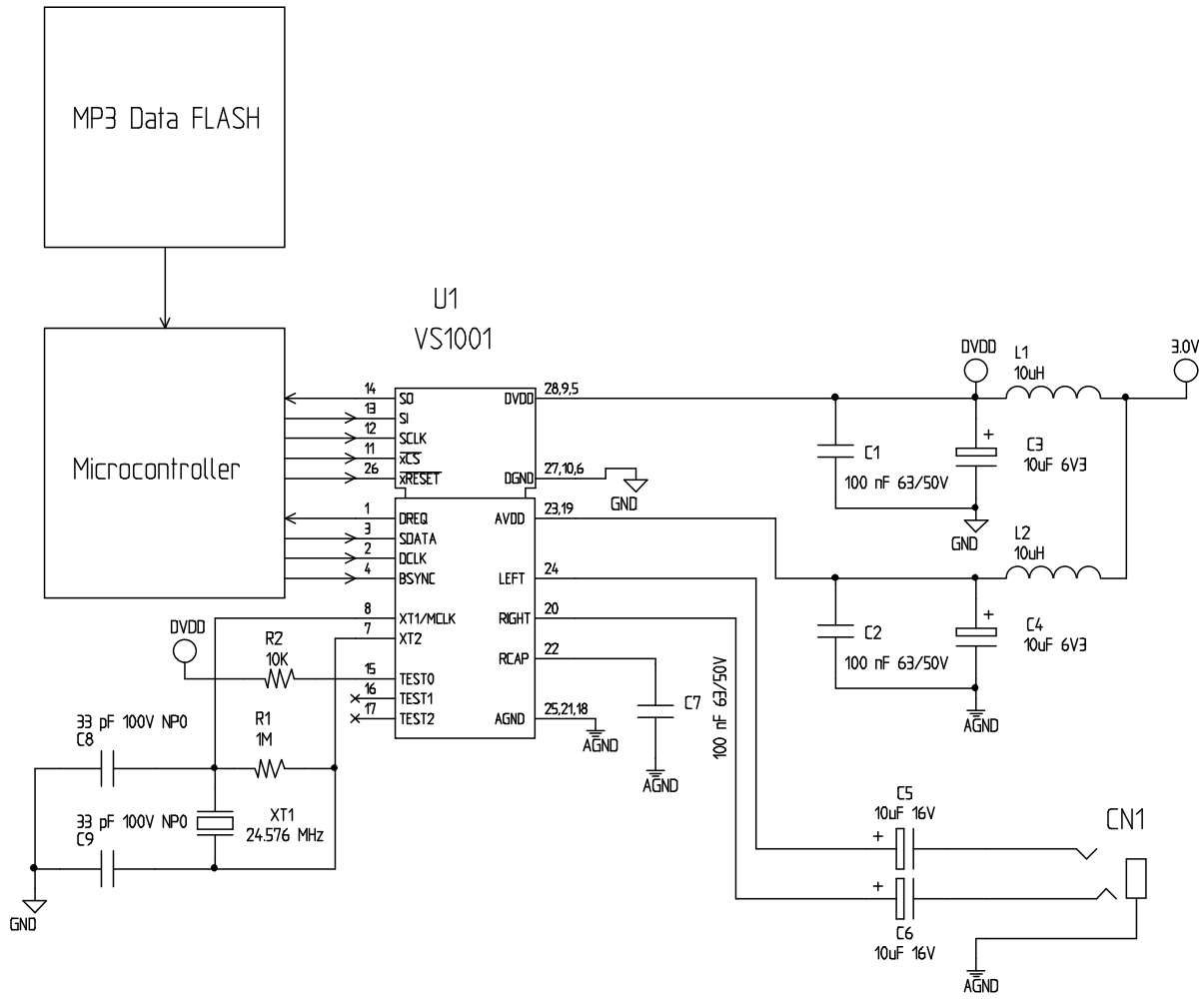


Figure 3: Typical Connection Diagram Using SOIC-28.

Ground buffer is not available for the SOIC-28 package; hence it is not used.

Note 1: With the implementation of the picture, bass frequency response below 300 Hz may not be adequate. To correct this, replace C5 and C6 with 100 μF capacitors.

Note 2: If running with full clock speed (i.e. if you don't use the internal clock doubler), replace C8 with a 10 pF capacitor and connect a 1 MΩ resistor in parallel with it. Also replace R1 with a 250 kΩ resistor.

4 Connection Diagram, BGA-49

In this connection diagram, a BGA-49 -packaged VS1001G is used. In this picture, ground buffer is active.

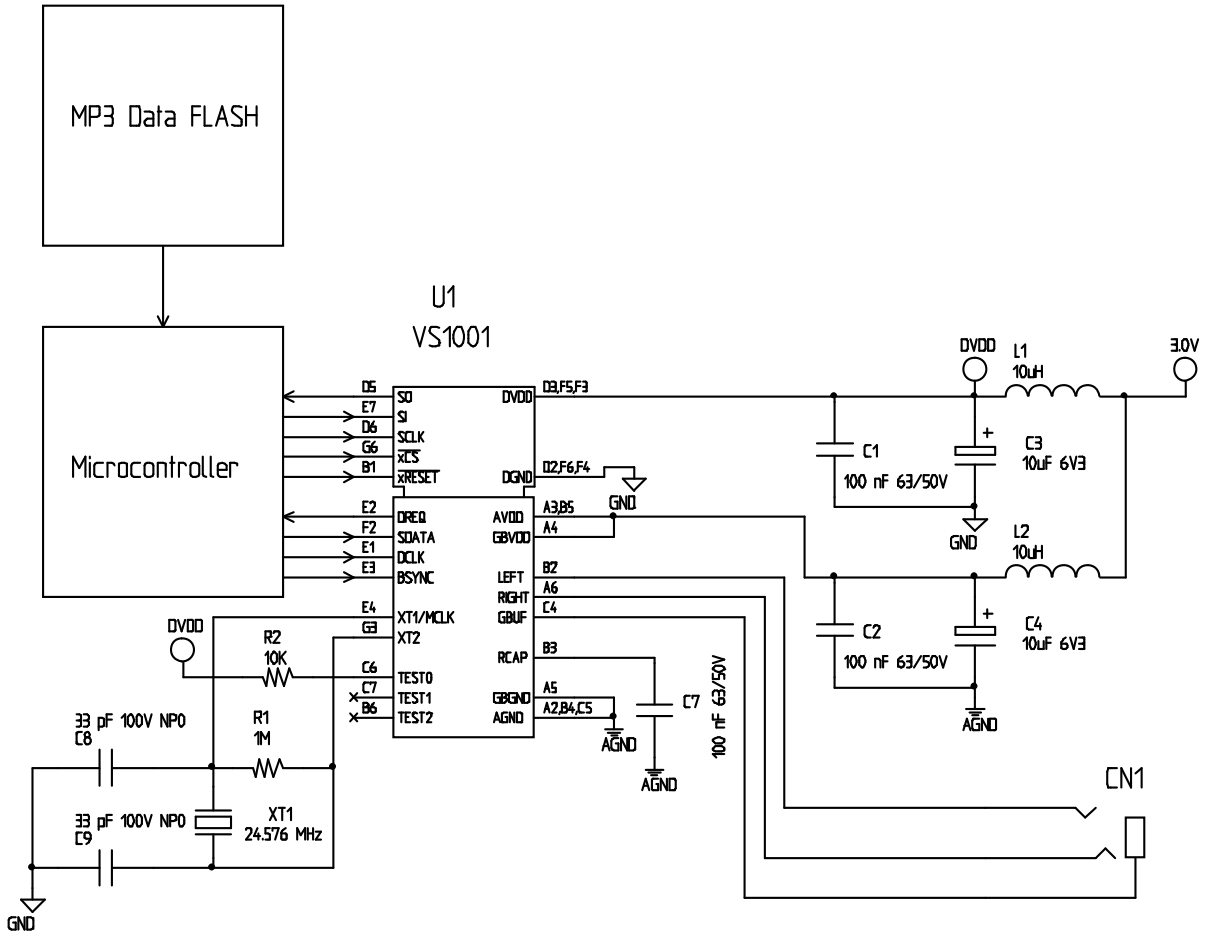


Figure 4: Typical Connection Diagram Using BGA-49.

Ground buffer GBUF can be used for common voltage (1.37 V) for earphones. This will eliminate the need for large isolation capacitors on line outputs, and thus the audio output pins from VS1001G may be connected directly to the earphone connector. If GBUF is not used, GBGND and GBVDD should not be connected.

Note: If running with full clock speed (i.e. if you don't use the internal clock doubler), replace C8 with a 10 pF capacitor and connect a 1 MΩ resistor in parallel with it. Also replace R1 with a 250 kΩ resistor.

5 SPI Buses

5.1 General

The SPI Bus - that was originally used in some Motorola devices - has been used for both VS1001G's Serial Data Interface SDI (C.f. Chapters 5.3 and 6.2) and Serial Control Interface SCI (C.f. Chapters 5.4 and 6.3).

5.2 SPI Bus Pin Descriptions

SDI Pin	SCI Pin	Description
-	XCS	Active low chip select input. A high level forces the serial interface into standby mode, ending the current operation. A high level also forces serial output (SO) to high impedance state. There is no chip select for SDI, which is always active.
DCLK	SCK	Serial clock input. The serial clock is also used internally as the master clock for the register interface. SCK can be gated or continuous. In either case, the first rising clock edge after XCS has gone low marks the first bit to be written (clock 0 in the following figures).
SDATA	SI	Serial input. SI is sampled on the rising SCK edge, if XCS is low.
-	SO	Serial output. In reads, data is shifted out on the falling SCK edge. In writes SO is at a high impedance state.

5.3 Serial Protocol for Serial Data Interface (SDI)

The serial data interface can operate in either master or slave mode. In master mode, VS1001G generates the DCLK signal, which can be selected to be either 512 or 1024 kHz. In slave mode, the DCLK signal is generated by an external circuit.

The data (SDATA signal) can be clocked in at either the rising or falling edge of the DCLK. (C.f. Chapter 6.4).

The VS1001G chip assumes its input to be byte-sychronized. I.e. the internal operation of the decoder does not search for byte synchronization of the frames from the data stream, but instead assumes the data to be correctly byte-aligned. The bytes can be transmitted either MSB or LSB first, depending of contents of SCI register MODE (C.f. Chapter 6.4).

To ensure correct byte-alignment of the input bitstream, the serial data interface has a BSYNC signal. The first DCLK sampling edge (rising or falling, depending on selected polarity), during which the BSYNC is high, marks the first bit of a byte (LSB, if LSB-first order is used, MSB, if MSB-first order is used). If BSYNC is not used, it must be tied to VCC externally and the master of the input serial interface must always sustain the correct byte-alignement. Using BSYNC is strongly recommended. For more details, look at the Application Notes in Chapter 12.

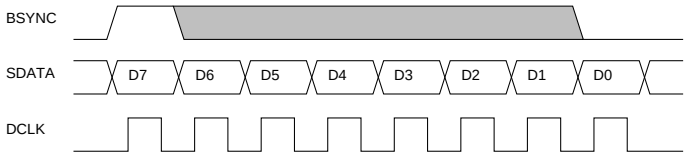


Figure 5: BSYNC Signal

The DREQ signal of the data interface is used in slave mode to signal if VS1001G's FIFO is capable of receiving more input data. If DREQ is high, VS1001G can take at least 32 bytes of data. When there is less than 32 bytes of free space, DREQ is turned low, and the sender should stop tranferring new data. Because of the 32-byte safety area, the sender may send upto 32 bytes of data at a time without checking the status of DREQ, making controlling VS1001G easier for low-speed microcontrollers.

5.4 Serial Protocol for Serial Command Interface (SCI)

5.4.1 General

The serial bus protocol for the Serial Command Interface SCI (c.f. Chapter 6.3) consists of an instruction byte, address byte and one 16-bit data word. Each read or write operation can read or write a single register.

The operation is specified by an 8-bit instruction opcode. The supported instructions are read and write. See table below.

Instruction		
Name	Opcode	Operation
READ	0000 0011	Read data
WRITE	0000 0010	Write data

Note: After using the Serial Command Interface, it is not allowed to send SCI or SDI data for 5 microseconds.

5.4.2 SCI Read

VS1001G registers are read by the following sequence. First, XCS line is pulled low to select the device. Then the READ opcode (0x3) is transmitted via the SI line followed by an 8-bit word address. After the address has been read in, any further data on SI is ignored. The 16-bit data corresponding to the received address will be shifted out onto the SO line.

XCS should be driven high after the data has been shifted out. In that case, the word address will be incremented and data corresponding to the next address will be shifted out. After the last word has been shifted out, XCS should be driven high to end the READ sequence.

Word read is shown in figure 6.

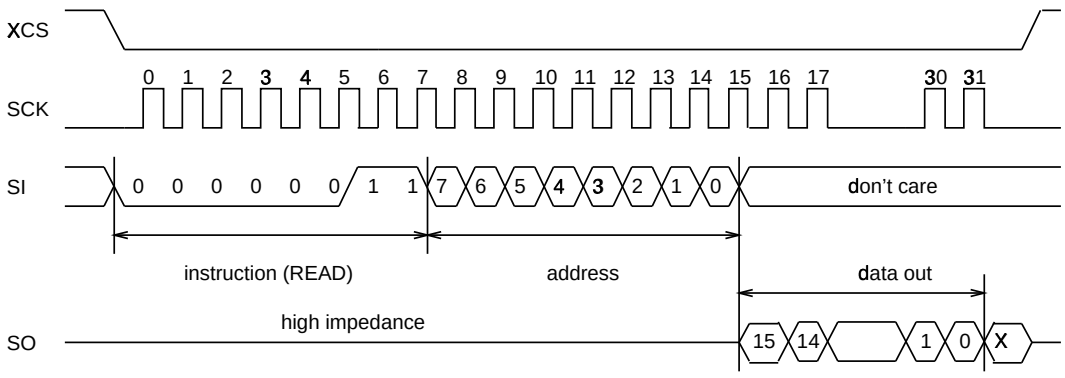


Figure 6: SCI Word Read

5.4.3 SCI Write

VS1001G registers are written by the following sequence. First, XCS line is pulled low to select the device. Then the WRITE opcode (0x2) is transmitted via the SI line followed by an 8-bit word address.

After the word has been shifted in, XCS should be pulled high to end the WRITE sequence. XCS low to high transition must occur after SCLK high to low transition corresponding to LSB of the last word.

Single word write is shown in figure 7.

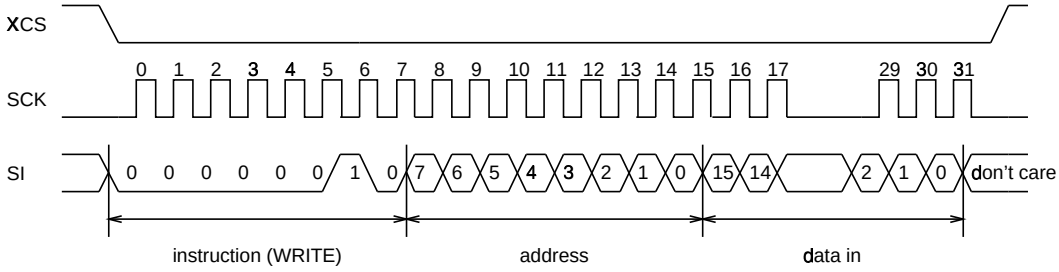


Figure 7: SCI Word Write

5.5 SPI Timing Diagram

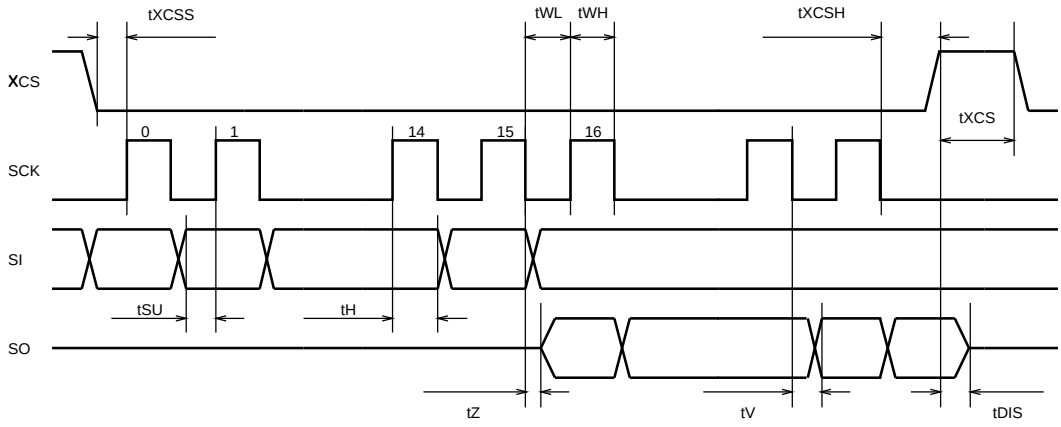


Figure 8: SPI Timing Diagram.

Symbol	Min	Max	Unit
tXCSS	5		ns
tSU	10		ns
tH	42		ns
tZ		42	ns
tWL	100		ns
tWH	100		ns
tV		42	ns
tXCSH	10		ns
tXCS	2		XTALI cycles
tDIS		1	XTALI cycles

Note, that as tXCS must be at least 2 clock cycles, the maximum safe speed for the SPI bus is 1/4 of VS1001G's clock speed.

6 Functional Description

6.1 Main Features

VS1001G is based on a proprietary digital signal processor, VS_DSP. It contains all the code and data memory needed for MPEG audio decoding, together with serial interfaces, a multirate stereo audio DAC and analog output amplifiers and filters.

VS1001G can play all MPEG 1 and 2 layer I, II and III files, as well as so-called MPEG 2.5 layer III extension files with all samplerates and bit-rates. In addition, variable bitrate (VBR) is also supported. With VBR, and depending on the song, near-cd quality can be achieved with approximately 100 kbits/s for stereo music sampled at 44100 Hz, whereas old encoders required 128 kbits/s for the same task. As both commercial and free (*http://www.lame.org/*) high-quality VBR encoders are nowadays widely available, MP3 format is getting better as it is maturing.

6.2 Serial Data Interface (SDI)

The serial data interface is meant for transferring compressed MPEG audio data.

Also several different tests may be activated through SDI as described in Chapter 7.

6.3 Serial Control Interface (SCI)

The serial control interface is compatible with the SPI-bus specification. Data transfers are always 16-bits. The VS1001G is controlled by writing and reading the registers of the interface.

The main controls of the control interface are:

- control of the operation mode
- uploading user programs
- access to header data
- status information
- access to decoded digital data
- feeding input data

6.4 SCI Registers

Name	Type	addr	Function
MODE	RW	0	mode control
STATUS	RW	1	status of VS1001G
INT_FCTLH	-	2	internal register
CLOCKF	RW	3	clock freq + doubler
SRATE	R	4	current sample rate
AUDATA	R	5	misc. audio data
WRAM	W	6	RAM write program
WRAMADDR	W	7	base address for RAM write
HDAT0	R	8	read header data
HDAT1	R	9	read header data
A1ADDR	RW	10	start address of application
VOL	RW	11	volume control
A1CTRL[x]	RW	12+x	3 application control registers

x = [0 .. 2]

All registers are filled with zeros at hardware reset.

6.4.1 MODE (RW)

MODE is used to control the operation of VS1001G.

bit	function		
MODE[0]	read WRAM	0 1	disabled enabled
MODE[1]	Not used		
MODE[2]	Soft reset	0 1	no reset reset
MODE[3]	Not used		
MODE[4]	powerdown	0 1	power on powerdown
MODE[5]	Not used		
MODE[6]	Not used		
MODE[7]	Not used		
MODE[8]	DCLK active edge	0 1	rising falling
MODE[9]	Byte order on serial input bus	0 1	MSB first MSB last
MODE[10]	input bus mode	0 1	slave master
MODE[11]	input bus clk when VS1001G master	0 1	512 kHz 1024 kHz

By setting Bit 2 to 1, the player is reset.

Bit 4 overrides any other: it turns VS1001G into powerdown mode, where the only operational part is the control bus.

Bit 8 defines the active edge of data clock for SDI.

Bit 9 defines the data order inside a byte for SDI. Bytes are, however, still sent in the default order.

Bit 10 sets input bus to master mode. Master mode has not been tested, and its use is not recommended.

Bit 11 sets the bus clock speed when VS1001G is the master.

6.4.2 STATUS (RW)

STATUS contains information on the current status of the VS1001G. Bits 1 and 0 are used to control analog output volume: 0 = -0 dB, 1 = -6 dB, 3 = -12 dB. Bit 2 is analog powerdown bit. When set to 1, analog is put to powerdown.

6.4.3 INT_FCNTLH (-)

INT_FCTLH is not a user-accessible register.

6.4.4 CLOCKF (RW)

CLOCKF is used to tell if the clock is running at something else than 24.576 MHz. The clock frequency is set in 2 kHz steps. Thus, the formula for calculating the correct value for this register is $ClockInHz/2000$. Values may be between 0..32767, although hardware limits the highest allowed speed. Also, with lower-than 24.576 MHz speeds all sample rates and bit-stream widths are no longer available. Setting the MSB of CLOCKF to 1 activates internal clock-doubling. A clock of upto 15 MHz may be doubled. If the clock doubler is activated, the 15 LSBs of the registers are calculated using the clock-doubled value. Note that this register must be set before baginning decoding MP3 data; otherwise the sample rate will not be set correctly.

Example 1: For a 26 MHz clock the value would be 13000.

Example 2: For a 13.5 MHz external clock and using internal clock-doubling for a 27 MHz internal frequency, the value would be 0x8000 + 13500 = 46268.

6.4.5 SRATE (R)

When decoding correct data, the audio sample rate can be found in SRATE as an unsigned integer.

6.4.6 AUDATA (R)

When decoding correct data, the current bitrate in kbits/s can be found in bits 8..0 of AUDATA. For a variable bitrate bitstream, the current bitstream width is displayed. Bits 14..9 are not in use and always set to 0. Bit 15 is 0 for mono data and 1 for stereo.

6.4.7 WRAM (W)

WRAM is used to upload application programs to program RAM. The start address must be initialized by writing to the WRAMADDR register prior to the first call of WRAM. value will be used. As 16 bits of data can be transferred with one WRAM write, and the program word is 32 bits, two consecutive writes are needed for each program word. The byte order is big-endian (i.e. MSBs first). After each full-word write, the internal pointer is autoincremented.

6.4.8 WRAMADDR (W)

WRAMADDR is used to set the program address for following WRAM writes. User program space is between addresses 4096 .. 5119 (with addresses 4096 .. 4111 being reserved by the system), but for writes through the WRAM mechanism, they are visible at addresses 4096 higher. Thus, if the programmer wish to write his application to address 4567, he should write 4567 + 4096 = 8663 to WRAMADDR.

6.4.9 HDAT0 and HDAT1 (R)

bit	function	value	explanation
HDAT1[15:5]	syncword	2047	stream valid
HDAT1[4:3]	ID	3	MPG 2.5 (1/4-rate)
		2	MPG 2.5 (1/4-rate)
		1	ISO 11172-3 1.0
		0	MPG 2.0 (1/2-rate)
HDAT1[2:1]	layer	3	I
		2	II
		1	III
		0	reserved
HDAT1[0]	protect bit	1	CRC protected
		0	No CRC
DAT0[15:12]	bitrate		c.f. 11172-3
HDAT0[11:10]	samplerate	3	reserved
		2	32/16/8 kHz
		1	48/24/12 kHz
		0	44/22/11 kHz
HDAT0[9]	pad bit	1	additional slot
		0	normal frame
HDAT0[8]	private bit		not defined
HDAT0[7:6]	mode	3	mono
		2	dual channel
		1	joint stereo
		0	stereo
HDAT0[5:4]	extension		c.f. 11172-3
HDAT0[3]	copyright	1	copyrighted
		0	free
HDAT0[2]	original	1	original
		0	copy
HDAT0[1:0]	emphasis	3	CCITT J.17
		2	reserved
		1	50/15 microsec
		0	none

When read, HDAT0 and HDAT1 contain header information that is extracted from MPEG stream being currently being decoded. Right after resetting VS1001G, 0 is automatically written to both registers, indicating no data has been found yet.

The “samplerate” field in HDAT0 is interpreted as follows: if the “ID” field in HDAT1 is ’1’, the highest sample rate is used. If “ID” is ’0’, half sample rate is used. For ’2’ and ’3’, the lowest sample rate is used.

Note, that the samplerate, stereo/mono and bitrate information can more easily be read from registers SRATE and AUDATA.

6.4.10 A1ADDR (RW)

A1ADDR indicates the start address of the application code written earlier through WRAMADDR and WRAM registers. If no application code is used, this register should not be initialized, or it should be initialized to zero.

6.4.11 VOL (RW)

VOL is a volume control for the player hardware. For each channel, a value in the range of 0 .. 255 may be defined to set its attenuation from the maximum volume level (in 0.5 dB steps). The left channel value is then multiplied by 256 and the values are added. Thus, maximum volume is 0 and total silence if 65535. Example: for a volume of -2.0 dB for the left channel and -3.5 dB for the right channel: (4*256) + 7 = 1031. Note, that at startup volume is set to full volume. Resetting the software does not reset the volume setting.

Note, that setting the volume to total silence (255 for both left and right channels), will turn analog power off. This will save power, but also cause a slight snap in the earphones. If you want to turn the volume off but don’t want this snap, turn the volume only to 254 for both channels (0xFEFE).

6.4.12 A1CTRL[x] (RW)

A1CTRL[x] -registers (x=[0 .. 2]) can be used to access the user’s application program.

6.5 Application Programs

There is 1 kWord (32-bit) of RAM memory for user code on the VS1001G -chip. This can be used for example to provide features like:

- Bass, treble (etc.) controls
- Channel mixing (stereo into mono)
- Digital equalizer

The loading of application programs is initiated by writing a base address to WRAMADDR register. The program is then loaded by writing data to the WRAM register.

Any programs have to be reloaded every time the chip loses its power supply. The application address does not have to be set every time after the system software has been reset.

The C prototype for the application function is as follows:

void (*applAddr) (s_int16 __i0 *data, s_int16 __a0 chan, s_int16 __a1 nSampl);

If $chan$ is either 1 or 2, the user may do in-place filtering of $data$. $chan$ is the number of channels, and $nSampl$ is the total number of samples to be handled. With a stereo signal, the left and right channel samples are interleaved. The program may trust in that $nSampl$ is always divisible by 4.

If $chan$ is 3, the volume has been set using SPI volume setting. In this case the volume is set automatically, but software is still provided with some data of the new volume setting: $nSampl$ contains the user selected volume setting, and d points to the left volume multiplier register. The right volume multiplier is in $d + 1$. The volume value is unsigned, and 32768 corresponds to a gain of 1.0 (e.g. $16384 -> 0.5$).

If $chan$ is 4..6, AICTRL[$chan$-4] has been called, and the user given value is in $nSampl$.

Volume control is placed after any user applications. Thus it is generally a better idea to only write filters that attenuate some frequencies and don't emphasize any. To compensate for the lower volume, main volume setting may be set up a few dB whenever tone control is activated.

An example of how to write an application that employs all the features mentioned here, may be found in separate application notes, available on VLSI Solution's Web pages.

6.6 Stereo Audio DAC

The decoded digital data is transformed into analog format by an 18-bit oversampling multi-bit sigma-delta DA-converter. The oversampled output is low-pass filtered by an on-chip analog filter. The output rate of the DA-converter is always 1/4 of the clock rate, or 128 times the highest usable samplerate. For instance for a 24.576 MHz clock, the DA-converter operates at 128x48 kHz, which is 6.144 MHz. If the input samplerate is other than 48 kHz, it is internally converted to 48 kHz by the DAC. This removes the need for complex PLL-based clocking schemes and still allows the use of several sample rates with one fixed master clock frequency.

The outputs can be separately muted by the user. If the output of the decoder is invalid or input data is not received fast enough, analog outputs are automatically muted. The analog outputs have buffers that are capable of driving 30Ω loads with a maximum of 50nF capacitance.

7 Operation

7.1 Clocking

The VS1001G chip operates typically on a single 24.576 MHz master clock frequency. This clock can be generated by an external circuitry (connected to pin XTALI) or by the internal clock chrystal interface (pins XTALI and XTALO). This clock is sufficient to support a high quality audio output for all the standard samplerates.

7.2 Powerdown

In powerdown mode the chip only monitors the control bus. The analog output drivers are turned off and the processor remains in hold-state.

7.3 Hardware Reset

When the XRESET -signal is driven low, VS1001G is reset and all the control registers and internal states are set to the initial values. XRESET-signal is asynchronous to any external clock. The reset mode doubles as a full-powerdown mode, where both digital and analog parts of VS1001G are in minimum power consumption stage, and where clocks are stopped.

After a hardware reset (or at power-up), set the basic software registers such as VOL for volume (and CLOCKF if the input clock is anything else than 24.576 MHz) before starting decoding.

7.4 Software Reset

Before first playback and between any two MP3 files, the decoder software has to be reset. This is done by activating bit 2 in SCI's MODE register (c.f. Chapter 6.4.1). Then wait for at least 2 μs, then look at DREQ. DREQ will stay down for at least 6000 clock cycles, which means an approximate 250 μs delay if VS1001G is run at 24.576 MHz. When DREQ goes up, write at least one zero to SDI. After this, you may contimue playback as usual.

7.5 Play/Decode

This is the normal operation mode of VS1001G. The SDI data is decoded. Decoded samples are converted to analog domain by the internal DAC, If there are errors in the decoding process, the error flags of SCI's HDAT0 and HDAT1 are set accordingly. In case there are serious errors in the input data, decoding is still continued, but the analog outputs are muted.

When there is no valid input for decoding, VS1001G goes into idle mode (lower power consumption

than during decoding) and actively monitors the serial data input for valid data. The data input does not need to be clocked (DCLK) when no data is sent.

The software needs to be reset between MPEG audio stream files. See for the Chapter “Testing” to see how it is done.

7.6 Sanity Checks

Although VS1001G checks extensively for bad MP3 streams, it may happen that it encounters a bitstream that makes the firmware’s recovery code fail. This may particularly happen during fast forward and fast backwards operations, where the data where the microcontroller lands the MP3 decoder may not be a valid header.

The microcontroller should keep a look at the data speeds VS1001G requires. If data input either stops completely (DREQ always inactive) for a whole second, or if VS1001G requires more than 60 KB data in any single second, it is the responsibility of the microcontroller to either reset the software. If that doesn’t help, a hardware reset should be issued.

7.7 Testing

There are several test modes in VS1001G, which allow the user to perform memory tests, SPI bus tests, and several different sine wave tests ranging from 250 Hz to 1500 Hz.

All tests are started in a similar way: if some MP3 has been decoded, the decoding has to be finished by sending 1024 zero bytes. This ensures that the decoder is looking for the next header and will not translate the testing commands to be valid MP3 data. If no MP3 data has been decoded since the last reset, this step doesn’t have to be taken.

Each test is started by sending a 4-byte special command sequence. The sequences are described below. Note, that after each special command, at least 4 zeros must be transmitted for the command to take effect.

7.7.1 Memory Test

Memory test mode is initialized by the sequence 0x4D 0xEA 0x6D 0x54. After this command (and its required 4 zeros), wait for 500000 clock cycles. The result can be read from the SPI register HDAT0, and ’one’ bits are interpreted as follows:

Bit(s)	Meaning
0	Good X ROM
1	Good Y ROM (low)
2	Good Y ROM (high)
3	Good Y RAM
4	Good X RAM
5	Good Instruction RAM (high)
6	Good Instruction RAM (low)
7	Unused

All tests are non-destructive and interrupts are disabled during testing. Thus, no user software or data is harmed by the tests.

Instruction ROM cannot be tested with software.

7.7.2 SCI Test

Sci test is initialized by the sequence: 0x53 0x70 0xEE n, where $n - 48$ is the register number to test. The content of the given register is read and copied to HDAT0. If the register to be tested is HDAT0, the result is copied to HDAT1.

7.7.3 Sine Test

Sine test is initialized by the sequence: 0x53 0xEF 0x6E n, where n (48..119) defines the sine test to use. If we define $FsIdx = (n - 48) mod 9$ and $FSin = (n - 48)/9$, the following tables may be used:

FsIdx	Fs
0	44100 Hz
1	48000 Hz
2	32000 Hz
3	22050 Hz
4	24000 Hz
5	16000 Hz
6	11025 Hz
7	12000 Hz
8	8000 Hz

FSin	Length of Sin
0	32.000 samples
1	16.000 samples
2	10.667 samples
3	8.000 samples
4	6.400 samples
5	5.333 samples
6	4.571 samples
7	4.000 samples

Example: Sine test is called with a test value of 62. 62-48 = 14, FsIdx = 5 and FSin = 1. From the tables we get the sample rate 16000 Hz, and the sine wave length, which is 16 samples. Thus, we'll get a 1 kHz voice.

To exit the sine test, send the sequence 0x45 0x78 0x69 0x74.

Note, that the sine test signals go through the digital volume control, so it is possible to test channels separately.

8 Clock Speeds

8.1 General

Using another clock speed than the default 24.576 MHz changes the behaviour of VS1001G in many ways. These are the maximum sample rate, the maximum decodable bitstream, and the maximum amount of allowed DSP effects.

8.2 Maximum Sample Rate

With a 24.576 MHz or higher clock, the maximum sample rate is 48000 Hz. For a lower clock, $48000 * ClockInMHz/24.576$ rounded downwards to the next allowed MP3 sample rate tells the maximum available sample rate. Thus, with a 15 MHz clock it is possible to decode playback at $29297 -> 24000$ Hz.

To put it another way: if VS1001G is run with a 26 MHz clock, the maximum playback frequency would be 50781 Hz, and all MPEG audio files can be played correctly. However, with a 24 MHz clock the maximum playback rate is only 46875 Hz. In this case, all sample rates upto 44100 Hz can be played withoput problems, but 48000 Hz is played at 46875 Hz, or in other words with a 2.5% speed error.

8.3 Maximum Decodable Bitstream

With a 24.576 MHz clock, the maximum decodable MP3 bitstream is 256 kbit/s. With a 28.5 MHz clock this can be brought upto 320 kbits/s, and thus covering all different MP3 bitstreams. For lower clocks, lower bitrates can be decoded. E.g. for a 22 kHz 128 kbit/s bitstream, 12.5 MHz is sufficient.

8.4 Maximum Amount of DSP Effects

With a 24.576 MHz clock there is some 7 MHz free processor time (minimum) for a 128 kbits/s 44.1 kHz bitstream. However, with 256 kbits/s at 48 kHz there is no constant free time: although the average load is at approx. 22.5 MHz, peaks need the whole available 24.576 MHz.

If it is desired to be able to play higher-width bitstreams at the same time with special effects or tone controls, a higher clock rate is required. With 26 MHz a simple tone control can be added to a 48 kHz 256 kbits/s bitstream, and with 30 MHz the same can be done to a full-width 48 kHz 320 kbits/s stream. With 40 MHz the programmer may add very complex DSP surround or bass-enhancement effects.

If a user wants to have a lower clock but still have DSP special effects, it may be a good idea to research the bitstream through SRATE and AUDATA registers. If the bitstream width and sample rates seem to be too high, turn DSP effects off using the A1ADDR register; otherwise turn them on.

9 Writing Software

9.1 When to Write Software

User software is required when a user wishes to add some own functionality like DSP effects or tone controls to VS1001G. Some tone controls are available from VLSI Solution, but if a user wishes to go further than that or use VS1001G in some unexpected way, this is how to do it.

However, most of the users of VS1001G don't need to worry about writing their own code, or this chapter.

9.2 The Processor Core

VS_DSP is a 16/32-bit DSP processor core that can very well also be used as an all-purpose processor. The VLSI Solution's free VSKIT Software Package contains all the tools and documentation needed to write, simulate and debug Assembly Language or ANSI C programs for the VS_DSP processor core.

The VSKIT Software Package is available on request from VLSI Solution.

9.3 User's Memory Map

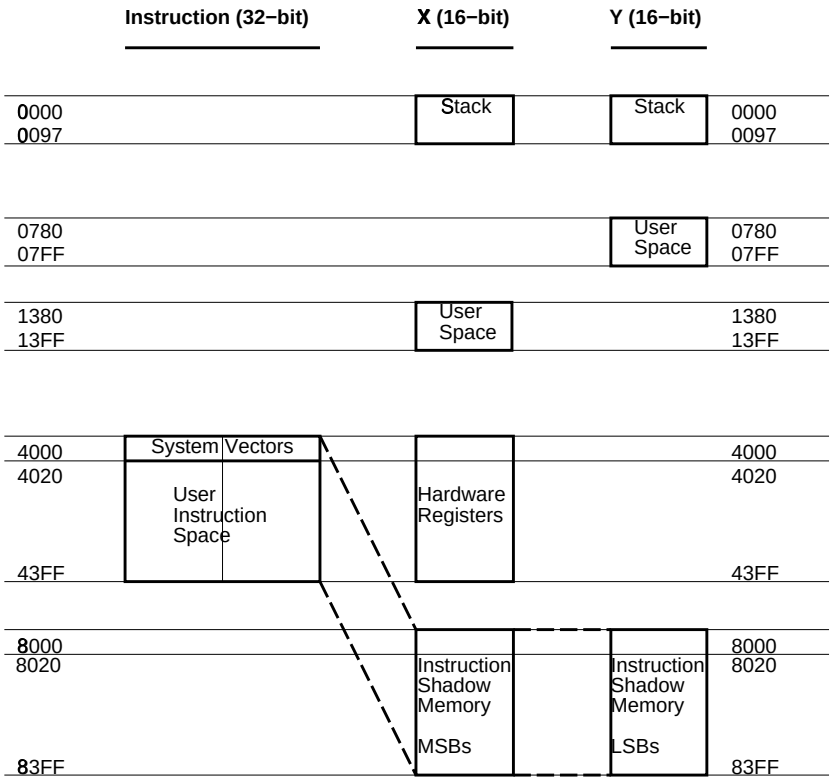


Figure 9: User's Memory Map

9.4 Hardware Registers

All hardware registers are located in X memory.

9.4.1 SCI Registers, 0x4000

All SCI registers described in Chapter 6.4 can be found here between 0x4000..0x40FF.

9.4.2 Serial Registers, 0x4100

SER_DATA (0x4100) contains the last data value read from the data bus. The LSB of SER_DREQ (0x4101) defines the status of the DREQ signal.

9.4.3 DAC Registers, 0x4200

DAC data should be written at each audio interrupt to DAC_LEFT (0x4200) and DAC_RIGHT (0x4201) as signed values. INT_FCTLL (0x4202) is not a user-serviceable register.

9.4.4 Interrupt Registers, 0x4300

INT_ENABLE (0x4300) controls the interrupts. Bit 0 switches the DAC interrupt on (1) and off (0), bit 1 controls the SPI interrupt, and bit 2 controls the DATA interrupt. It may take upto 6 clock cycles before changing this register has any effect.

By writing any value to INT_GLOB_DIS (0x4301) adds one to the interrupt counter and effectively disables all interrupts. It may take upto 6 clock cycles before writing this register has any effect.

Writing any value to INT_GLOB_ENA (0x4302) subtracts one from the interrupt counter. If the interrupt counter becomes zero, interrupts selected with INT_ENABLE are restored. An interrupt routine should always write to this register as the last thing it does, because interrupts automatically add one to the interrupt counter, but subtracting it back to its initial value is on the responsibility of the user. It may take upto 6 clock cycles before writing this register has any effect.

By reading INT_COUNTER (0x4303) the user may check if the interrupt counter is correct or not. If the register is not 0, interrupts are disabled. This register may not be written to.

9.5 System Vector Tags

The System Vector Tags are tags that may be replaced by the user to take control over several decoder functions.

9.5.1 AudioInt, 0x4000..0x4001

Normally contains the code:

```
j dac_int
stx mr1,(i6)+1  ; sty i7,(i6)
```

The user may, at will, replase the first instruction with either a j or $jmpi$ command to gain control over the audio interrupt. It is not recommended to change the instruction at 0x4001.

9.5.2 SpiInt, 0x4002..0x4003

Normally contains the code:

```
j spi_int
stx mr1,(i6)+1  ; sty i7,(i6)
```

The user may, at will, replase the first address with either a j or jmpi command to gain control over the spi interrupt. It is not recommended to change the instruction at 0x4003.

9.5.3 DataInt, 0x4004..0x4005

Normally contains the code:

```
j data_int
stx mr1,(i6)+1  ; sty i7,(i6)
```

The user may, at will, replase the first address with either a j or jmpi command to gain control over the MP3 data interrupt. It is not recommended to change the instruction at 0x4001.

9.5.4 UserCodec, 0x4008..0x4009

Normally contains the code:

```
jr
nop
```

If the user wants to take control away from the standard decoder, the first instruction should be replaced with an appropriate jump command to user's own code.

Unless the user is feeding MP3 data at the same time, the system activates the user program in less than 1 ms. After this, the user should steal interrupt vectors from the system, and then insert user programs.

9.6 System Vector Functions

The System Vector Functions are pointers to some some functions that the user may call to help implementing his own applications.

9.6.1 WriteIRam(), 0x4010

void WriteIRam(register __i0 u_int16 *addr, register __a1 msW, register __a0 lsW);

This is the only supported way to write to the User Instruction RAM. This is because Instruction RAM cannot be written when program control is in RAM. Thus, the actual implementation of this function is in ROM, and here is simply a tag to that routine.

Note, that Instruction RAM is shadowed 0x4000 addresses higher in the X and Y RAMs. Thus, if you want to write to instruction address 0x4020, $addr$ must be 0x4020 + 0x4000 = 0x8020.

9.6.2 ReadIRam(), 0x4011

u_int32 ReadIRam(register __i0 u_int16 *addr);

This is the only supported way to read from the User Instruction RAM. This is because Instruction RAM cannot be read when program control is in RAM. Thus, the actual implementation of this function is in ROM, and here is simply a tag to that routine.

A1 contains the MSBs and a0 the LSBs of the result.

Note, that Instruction RAM is shadowed 0x4000 addresses higher in the X and Y RAMs. Thus, if you want to read from instruction address 0x4020, $addr$ must be 0x4020 + 0x4000 = 0x8020.

9.6.3 DataWords(), 0x4012

u_int16 DataWords(void);

If the user has taken over the normal operation of the system by switching the pointer in UserCodec to point to his own code, he may read data from the Data Interface through this and the following two functions. This function returns the number of data words (each containing two bytes of data) that can be

read. If there is not enough data available, data acquisition functions GetDataByte() and GetDataWords() may NOT be called!

9.6.4 GetDataByte(), 0x4013

u_int16 GetDataByte(void);

Reads and returns one data byte from the Data Interface.

Before calling this function, always check first that there are at least 1 word waiting with function DataWords().

9.6.5 GetDataWords(), 0x4014

void GetDataWords(register __i0 u_int16 *d, register __a0 u_int16 n);

Read n data byte pairs and copy them in big-endian format (first byte to MSBs) to d.

Before calling this function, always check first that there are at least 1+n words waiting with function DataWords().

10 Application Note: Quick Startup

10.1 Overview

This chapter is meant to be read after the first device using VS1001G has been built. In this chapter the first steps to get VS1001G up and running are discussed. In addition to this chapter, also Chapter 11 is useful for the first steps.

These tests have been meant to be run one at a time, and the next test is to be done only after the previous one has been passed.

10.2 Seeing If Firmware Wakes Up

When VS1001G is taken out of hardware reset (xRESET is turned to 1), DREQ should turn to 0 after approx. 4096 clock cycles, if it hasn't been down all the time. After yet another 6000 clock cycles, DREQ should turn to 1. If DREQ doesn't change its mode as described, the firmware is not functioning properly.

10.3 Writing to SCI

The first test that does anything useful is to try to write to the SCI bus. A good test is to try to switch the volume setting from powersave mode to full volume. This will cause slight snapping sounds, which can be checked with the earphones.

The following algorithm can be used to test SPI this way:

- Activate xCS by setting it to 0.
- Set full volume by writing four SPI bytes: 0x2, 0xB, 0x0, 0x0
- Finish SCI cycle setting xCS to 1.
- Wait for example 500 ms.
- Activate xCS by setting it to 0.
- Set analog powerdown by writing four SPI bytes: 0x2, 0xB, 0xFF, 0xFF
- Wait for example 500 ms.
- Finish SCI cycle setting xCS to 1.
- Repeat these steps indefinitely.

10.4 Reading from SCI

Now that we know writing to SCI works, we can try reading from it. Write some recognizable pattern to the volume register, like 0xA2F5 (this is a good pattern, because it effectively tests both the MSB and LSB). Then, do as follows:

- Activate xCS by setting it to 0.
- Read the volume register as described in 11.3.3.
- Check, if the result is 0xA2F5.
- Finish SCI cycle setting xCS to 1.

10.5 Writing to SDI

Writing to the actual MP3 data SDI bus is most easily tested with the special VS1001G test header, like sine test (c.f. Chapter 7.7.3).

- Activate the sine test by writing the following eight bytes to SDI: 0x53 0xEF 0x6E 0x30, 0, 0, 0, 0. Remember to take care of BSYNC properly.
- Wait for example 500 ms.
- Deactivate the sine test by writing to SDI: 0x45 0x78 0x69 0x74.
- Wait for example 500 ms.
- Repeat these steps indefinitely.

If you can hear a half-second beep every second, you have properly entered and exited the sine test mode.

Note, that unless you atteanuate volume at least -20 dB from full volume, the sine tone will be very loud. It is thus not recommended to keep your headphones on while conducting this test.

11 Application Note: Saving I/O Pins

11.1 Overview

As can be seen from Figures 3 and 4, connecting VS1001G to a microcontroller requires 8 pins. This can, however be reduced to 7, or even 6 if VS1001G is the only SPI device connected to the microcontroller. A six pin minimal configuration is presented in Figure 10.

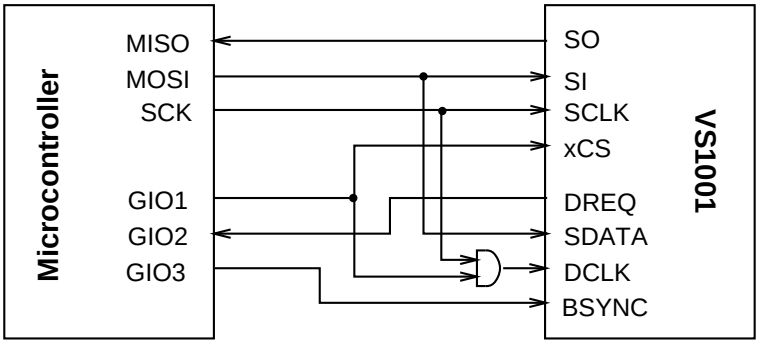


Figure 10: 6-Pin VS1001G Connection.

One pin can still be saved if BSYNC is connected permanently to VDD and it can be trusted that SDI bit sync is never lost.

11.2 Prerequirements

For this scheme to work, the following start conditions apply:

- MISO and GIO2 must be set to inputs, all other pins as outputs.
- SPI clocks have to be set so that when an SPI operation is not active, clock is set low, as in the oscilloscopes picture in Chapter 11.4.
- If there is no SPI bus available, but the microcontroller is very fast, MISO, MOSI and SCK may be implemented with general I/O pins.

11.3 Using the Connection

11.3.1 Selecting the Right Chip

In this connection, GIO1 acts as a bus select. When GIO1 is set to 1, xCS is inactive, and SCK is let through the AND gate to DCLK. Thus, now data is sent to the SDI bus. When GIO1 is set to 0, xCS activates, and DCLK stops because of the AND gate. Thus, data now goes only to the SCI bus.

If other SPI devices in addition to VS1001G is to be connected to the microcontroller, at least one more select pin is needed.

11.3.2 Sending SCI Data

Let's say we would like to set the volume to -2 dB for the left channel and to -3.5 dB to the right channel. Thus, we would like to write the number 0x0407 to the VOL register.

- Activate the SCI interface by setting GIO1 to 0
- Write the four bytes (0x2, 0xb, 0x4, 0x7) to SCI.
- Finish the SCI transaction by setting GIO1 to 1

11.3.3 Receiving SCI Data

Let's say we would like to read the contents of the VOL register.

- Activate the SCI interface by setting GIO1 to 0
- Write two bytes (0x3, 0xb) to SCI.
- Write one byte with any contents to SCI. Read the microcontroller's MISO register, which now contains 8 MSBs of the result.
- Write one byte with any contents to SCI. Read the microcontroller's MISO register, which now contains 8 LSBs of the result.
- Combine the MSBs and LSBs to form a 16-bit word.
- Finish the SCI transaction by setting GIO1 to 1

11.3.4 Sending SDI/MP3 Data

The following algorithm may be used to send a 32-byte or smaller MP3 data chunk to VS1001G:

- Wait until DREQ is 1.
- Activate DCLK by setting GIO to 1.
- For each byte, do the following:
  - Set BSYNC to 1.
  - Activize SPI transfer.
  - Wait until you know that at least the first bit of data has been sent, and the last bit has not been sent. Although VS1001G can survive several too long BSYNCs, it may be a good idea to disable interrupts to make sure the delay always is the same length.
  - Set BSYNC to 0.
  - Wait for microcontroller's SPI cycle to finish.

11.4 SDI in Oscilloscope Pictures

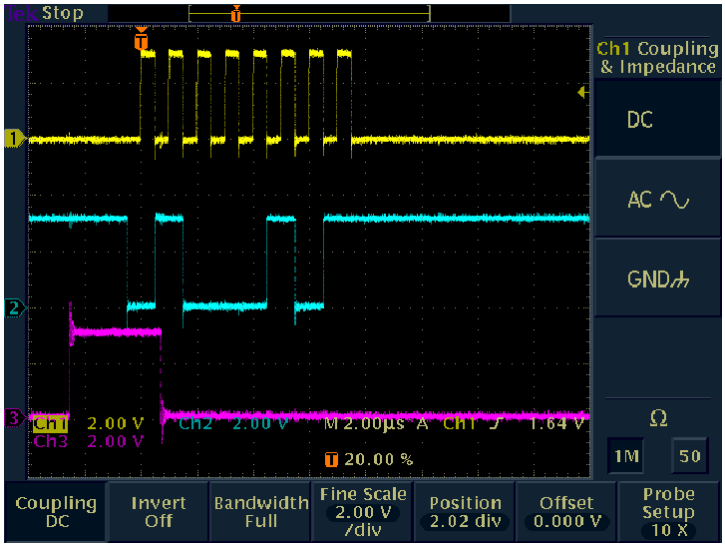


Figure 11: Oscilloscope Picture: Sending One SDI Byte

In Figure 11 sending of one byte to SDI is presented. The three signals from top to bottom are DCLK (yellow), SDATA (cyan), and BSYNC (magenta).

BSYNC is first turned active. Then data is sent one bit at a time, and the data is read by VS1001G at each rising edge. The number to be sent is 0b01000101, or 0x45. The clock used is 1 MHz, because that is the highest this particular microcontroller can offer when running at 4 MHz.

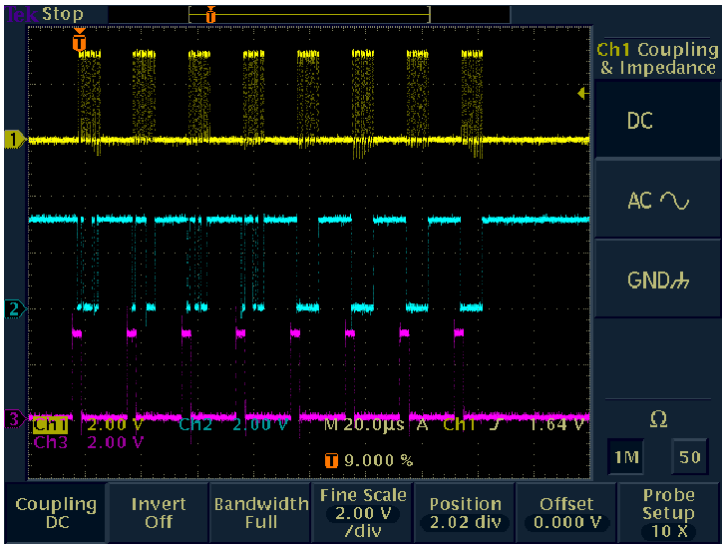


Figure 12: Oscilloscope Picture: Sending Eight SDI Bytes

In the other example ocilloscope picture, Figure 12, sending a whole SDI test command with padding zeroes is demonstrated. Here eight bytes are transmitted, and the bytes are 0x45 0x78, 0x69, 0x74, 0x0, 0x0, 0x0, 0x0. This is the sequence to exit sine test (C.f. Chapter 7.7.3). Note, that the delay between consecutive byte writes are caused by the slowness of the microcontroller C code and are not needed by VS1001G.

12 Application Note: MP3 Player

12.1 Overview

In this chapter, an MP3 player based on VS1001G is presented. The full schematics, as well as software is presented to make the device operate with a common MultiMediaCard through its SPI mode. The system software for the microcontroller is available from *http://www.vlsi.fi/software/vs1001.html* beginning from late March 2001.

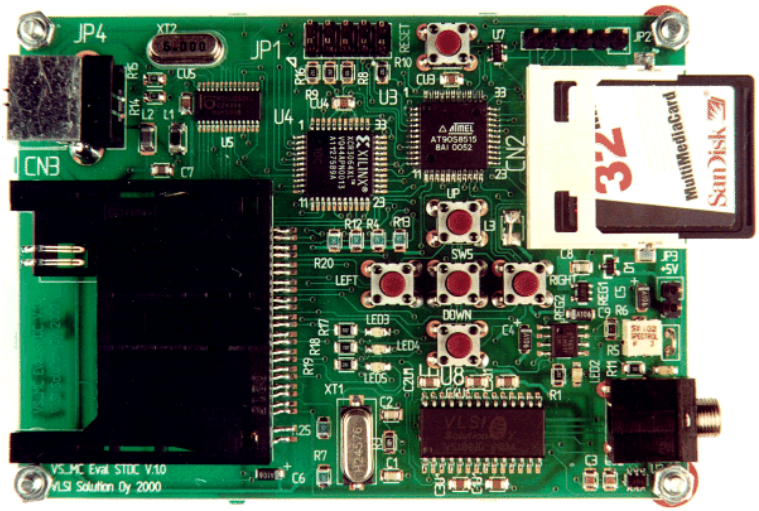


Figure 13: MP3 Evaluation Player.

12.2 Main Components

- Atmel 8-bit microcontroller AT90S8515 (8 KB program FLASH, SPI, etc)
- VS1001G
- SanDisk 32 MByte MultiMediaCard (any brand with SPI support will do)
- Xilinx XCR3064XL Programmable Logic Unit
- Philips PDIUSBD12 USB Controller
- 2 Leds for output
- 5 keys for input

The schematics also allow for on-board parallel FLASH memory as well as a Smart Media Card, but those features have not been tested.

12.3 Schematics for Standalone Unit

In Figures 14 and 15 are the full schematics of the standalone MP3 player. Note that many of the devices are not actually used, but they are there just for testing. Note also, that in page 1 both normal and clock-doubled clock connections are shown. The user needs to implement only one of these two, depending on the clock to be used. See Chapters 3 and 4 for further details.

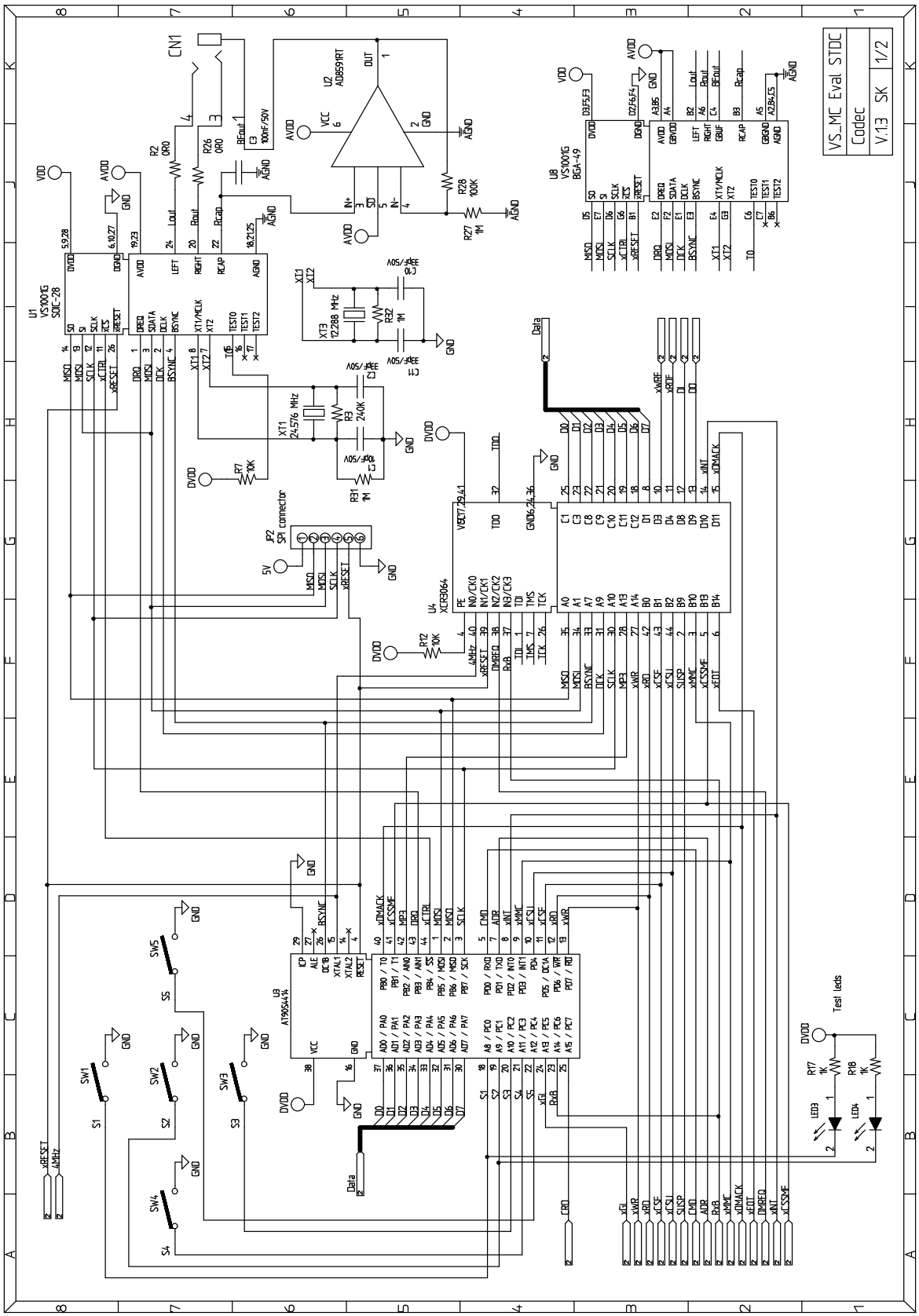


Figure 14: Schematic for Standalone MP3 Player, Page 1/2

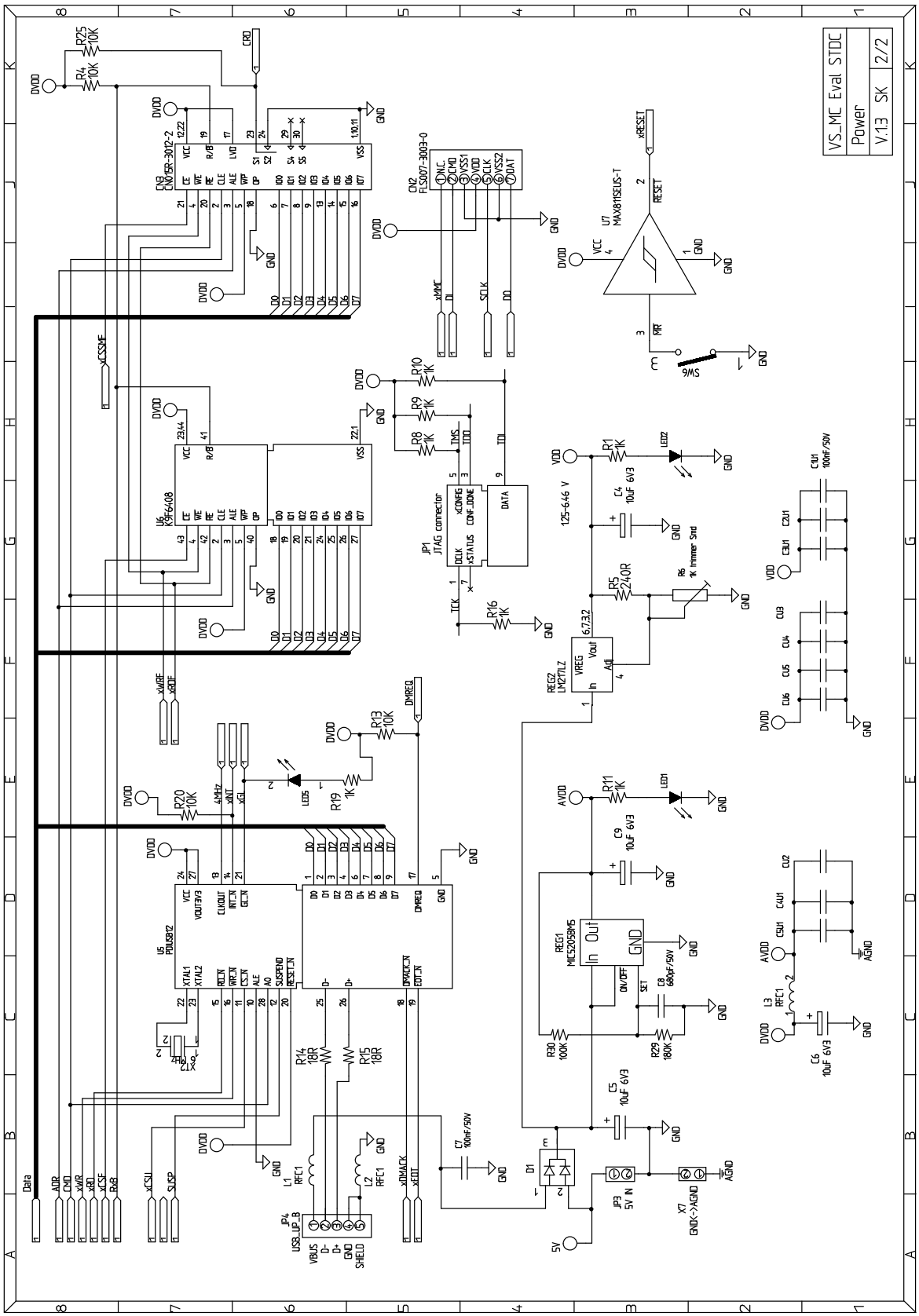


Figure 15: Schematic for Standalone MP3 Player, Page 2/2

12.4 Xilinx Configuration

In Figure 16 the full Xilinx configuration for the standalone player is displayed. The most complex part of it is the automatic BSYNC generation. This needn't, however, be generated in such a complex way. In many cases (including this one had the system not been built in the way it was), BSYNC may be generated with an I/O pin of the microcontroller.

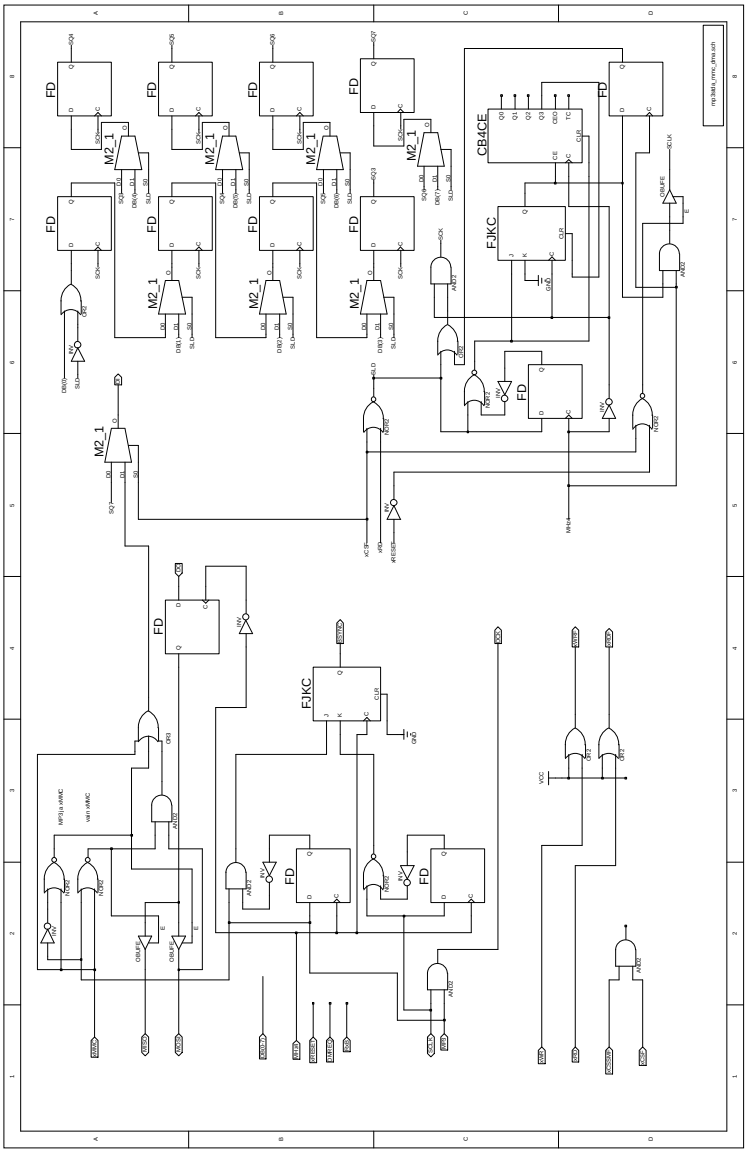


Figure 16: Xilinx Internal Configuration for Standalone MP3 Player

13 Contact Information

VLSI Solution Oy
Hermiankatu 6-8 C
FIN-33720 Tampere
FINLAND
Fax +358-3-316 5220
Phone +358-3-316 5230
Email sales@vlsi.fi
http://www.vlsi.fi/