Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Freescale Semiconductor
Advance Information

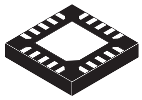
Document Number: SGTL5000
Rev.4.0,11/2011

RoHS

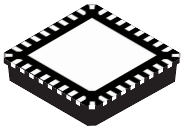
Low Power Stereo Codec with Headphone Amp

SGTL5000

AUDIO CODEC



PB-FREE
98ARE10742D
20-PIN QFN



PB-FREE
98ARE10739D
32-PIN QFN

ORDERING INFORMATION

Device	Temperature Range (T_A)	Package
SGTL5000XNLA3/R2	-40°C to 85°C	20 QFN
SGTL5000XNAA3/R2		32 QFN

The SGTL5000 is a Low Power Stereo Codec with Headphone Amp from Freescale, and is designed to provide a complete audio solution for portable products needing line-in, mic-in, line-out, headphone-out, and digital I/O. Deriving it's architecture from best in class, Freescale integrated products that are currently on the market. The SGTL5000 is able to achieve ultra low power with very high performance and functionality, all in one of the smallest footprints available. Target markets include portable media players, GPS units, and smart phones. Features such as capless headphone design and an internal PLL, help lower overall system cost.

Features

Analog Inputs

- Stereo Line In - Support for external analog input
- Stereo Line In - Codec bypass for low power
- MIC bias provided (5.0 x 5.0 mm QFN, 3.0 x 3.0 mm QFN TA2)
- Programmable MIC gain
- ADC - 85 dB SNR (-60 dB input) and -73 dB THD+N (VDDA = 1.8 V)

Analog Outputs

- HP Output - Capless design
- HP Output - 45 mW max into 16 ohm load @ 3.3 V
- HP Output - 100 dB SNR (-60 dB input) and -80 dB THD+N (V_{DDA} = 1.8 V, 16 ohm load, DAC to headphone)
- Line Out - 100 dB SNR (-60 dB input) and -85 dB THD+N (V_{DDIO} = 3.3 V)

Digital I/O

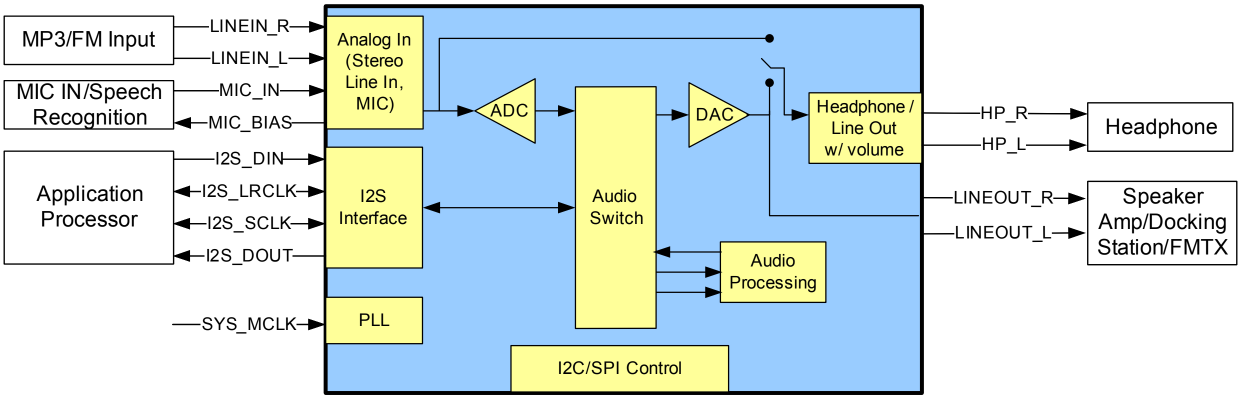
- I^2S port to allow routing to Application Processor

Integrated Digital Processing

- Freescale Surround, Freescale Bass, tone control/ parametric equalizer/graphic equalizer Clocking/Control
- PLL allows input of an 8.0 MHz to 27 MHz system clock - Standard audio clocks are derived from PLL

Power Supplies

- Designed to operate from 1.62 to 3.6 volts



Note: Only I^2C is supported in the 3.0 mm x 3.0 mm 20-pin QFN package option.

Figure 1. SGTL5000 Simplified Application Diagram

* This document contains certain information on a new product.
Specifications and information herein are subject to change without notice.

INTERNAL BLOCK DIAGRAM

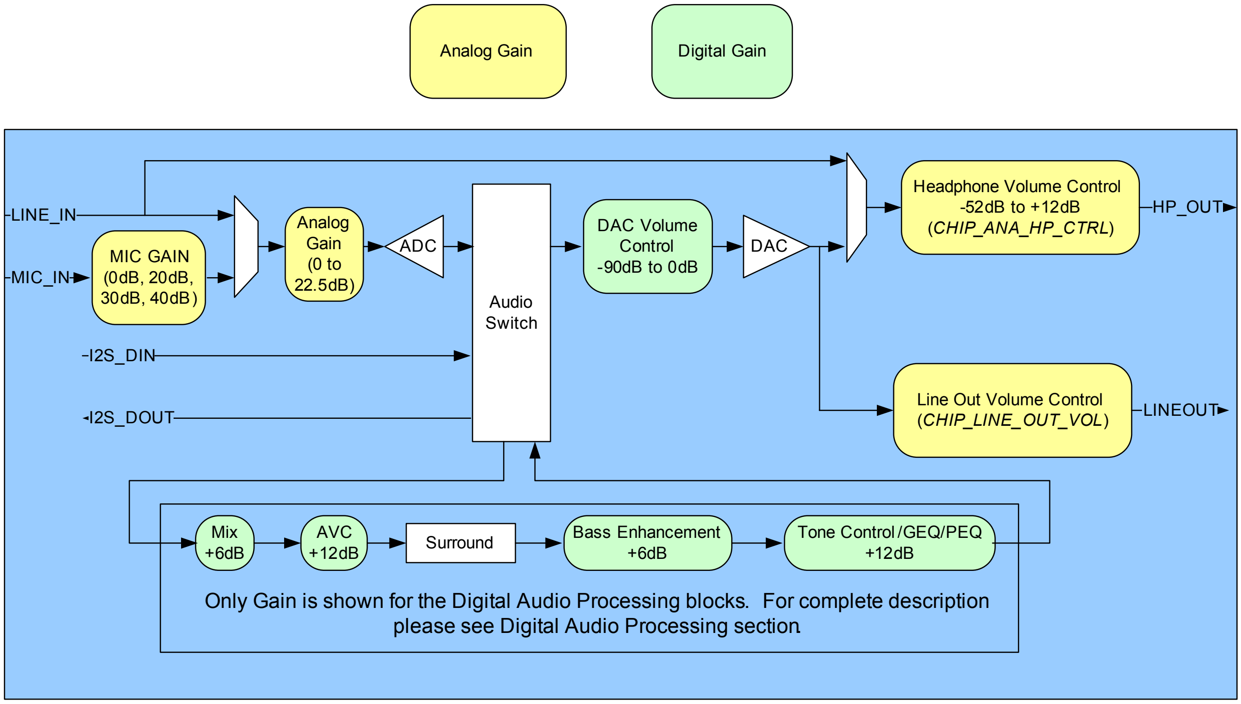
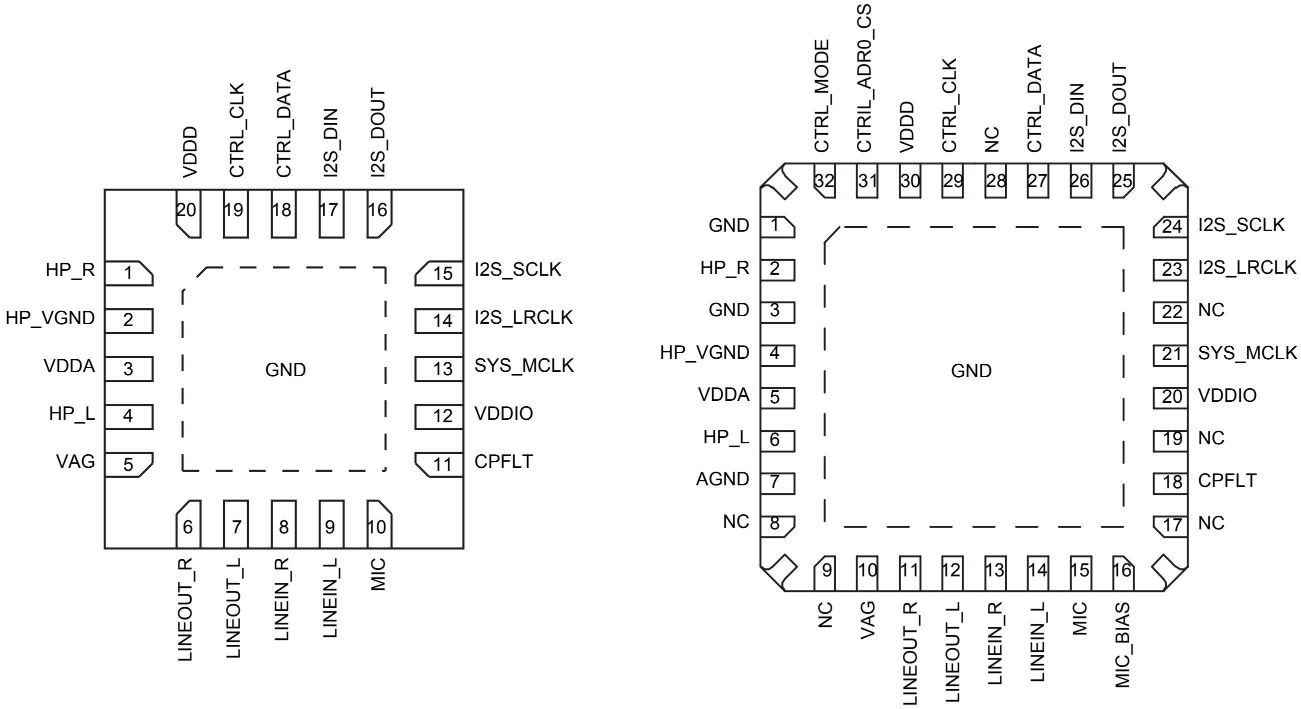

Analog Gain
Digital Gain
LINE_IN
MIC_IN
MIC GAIN (0dB, 20dB, 30dB, 40dB)
Analog Gain (0 to 22.5dB)
ADC
Audio Switch
DAC Volume Control -90dB to 0dB
DAC
Headphone Volume Control -52dB to +12dB (CHIP_ANA_HP_CTRL)
HP_OUT
Line Out Volume Control (CHIP_LINE_OUT_VOL)
LINEOUT
I2S_DIN
I2S_DOUT
Mix +6dB
AVC +12dB
Surround
Bass Enhancement +6dB
Tone Control/GEQ/PEQ +12dB
Only Gain is shown for the Digital Audio Processing blocks. For complete description please see Digital Audio Processing section.

Figure 2. SGTL5000 Simplified Internal Block Diagram

PIN CONNECTIONS



20 QFN
Transparent Top View

32 QFN
Transparent Top View

Figure 3. SGTL5000 Pin Connections

Table 1. SGTL5000 Pin Definitions

A functional description can be found in Functional Description, beginning on page 11.

20 Pin QFN	32 Pin QFN	Pin Name	Pin Function	Formal Name	Definition
1	2	HP_R	Analog	Right headphone output	
2	4	HP_VGND	Analog	Headphone virtual ground	Use the widest, shortest trace possible for the HP_VGND
3	5	VDDA	Power	Analog voltage	
4	6	HP_L	Analog	Left headphone output	
-	7	AGND	Analog Ground	Ground	
-	8, 9, 17, 19, 22, 28	NC	No Connect		
5	10	VAG	Analog	DAC VAG filter	
6	11	LINEOUT_R	Analog	Right line out	
7	12	LINEOUT_L	Analog	Left line out	
8	13	LINEIN_R	Analog	Right line in	
9	14	LINEIN_L	Analog	Left line in	
10	15	MIC	Analog	Microphone input	

Table 1. SGTL5000 Pin Definitions (continued)

A functional description can be found in Functional Description, beginning on page 11.

20 Pin QFN	32 Pin QFN	Pin Name	Pin Function	Formal Name	Definition
-	16	MIC_BIAS	Analog	Mic bias	
11	18	CPFILT	Analog	Charge Pump Filter	
12	20	VDDIO	Power	Digital I/O voltage	
13	21	SYS_MCLK	Digital	System master clock	
14	23	I2S_LRCLK	Digital	I^2S frame clock	
15	24	I2S_SCLK	Digital	I^2S bit clock	
16	25	I2S_DOUT	Digital	I^2S data output	
17	26	I2S_DIN	Digital	I^2S data input	
18	27	CTRL_DATA	Digital	I^2C Mode: Serial Data (SDA); SPI Mode: Serial Data Input (MOSI)	
19	29	CTRL_CLK	Digital	I^2C Mode: Serial Clock (SCL); SPI Mode: Serial Clock (SCK)	
20	30	VDDD	Digital	Digital voltage	
-	31	CTRL_AD0_CS	Digital	I^2C Mode: I^2C Address Select 0; SPI Mode: SPI Chip Select	
-	32	CTRL_MODE	Digital	Mode select for I^2C or SPI; When pulled low the control mode is I^2C, when pulled high the control mode is SPI	
PAD	1, 3, 4, PAD	GND	Ground	Ground	The PAD should be soldered to ground. This is a suggestion for mechanical stability but is not required electrically. Star the ground pins of the chip, VAG ground, and all analog inputs/outputs to a single point, then to the ground plane.

ELECTRICAL CHARACTERISTICS

MAXIMUM RATINGS

Table 2. Maximum Ratings

Exceeding the absolute maximum ratings shown in the following table could cause permanent damage to the part and is not recommended. Normal operation is not guaranteed at the absolute maximum ratings and extended exposure could affect long term reliability.

Ratings	Symbol	Value	Unit
ELECTRICAL RATINGS			
Maximum Digital Voltage	V_{DDD}	1.98	V
Maximum Digital I/O Voltage	V_{DDIO}	3.6	V
Maximum Analog Supply Voltage	V_{DDA}	3.6	V
Maximum voltage on any digital input		GND-0.3 to V_{DDIO}+0.3	V
Maximum voltage on any analog input		GND-0.3 to V_{DDA}+0.3	V
RECOMMENDED OPERATING CONDITIONS			
Digital Voltage (If supplied externally)	V_{DDD}	1.1 to 2.0	V
Digital I/O Voltage	V_{DDIO}	1.62 to 3.6	V
Analog Supply Voltage	V_{DDA}	1.62 to 3.6	V
THERMAL RATINGS			
Storage Temperature	T_{STG}	- 55 to 125	°C
Operating Temperature Ambient	T_A	-40 to 85	°C

STATIC ELECTRICAL CHARACTERISTICS

Table 3. Audio Performance 1

Test Conditions unless otherwise noted: V_{DDIO} = 1.8 V, V_{DDA} = 1.8 V, T_A = 25°C, Slave mode, f_S = 48 kHz, MCLK = 256 f_S, 24 bit input.

Characteristic	Symbol	Min	Typ	Max	Unit
AUDIO PERFORMANCE					
Line In Input Level		-	0.75	-	V_{RMS}
Line In Input Impedance		10	-	-	kOhm
LINE IN -> ADC -> I^2S OUT					
SNR (-60 dB input)		-	85	-	dB
THD+N		-	-70	-	dB
Frequency Response		-	±0.11	-	dB
Channel Separation		-	79	-	dB
LINE IN -> HEADPHONE_LINEOUT (CODEC BYPASS MODE)					
SNR (-60 dB input)		-	98	-	dB
THD+N (10 kOhm load)		-	-87	-	dB
THD+N (16 Ohm load)		-	-87	-	dB
Frequency Response		-	±0.05	-	dB
Channel Separation (1.0 kHz)			82		dB
I^2S IN -> DAC -> LINE OUT					
Output Level		-	0.6	-	V_{RMS}
SNR (-60 dB input)		-	95	-	dB
THD+N		-	-85	-	dB
Frequency Response		-	±0.12	-	dB
I^2S IN -> DAC -> HEADPHONE OUT - 16 OHM LOAD					
Output Power		-	17	-	mW
SNR (-60 dB input)		-	100	-	dB
THD+N		-	-80	-	dB
Frequency Response		-	±0.12	-	dB
I^2S IN -> DAC -> HEADPHONE OUT - 32 OHM LOAD					
Output Power		-	10	-	mW
SNR (-60 dB input)		-	95	-	dB
THD+N		-	-86	-	dB
Frequency Response		-	±0.11	-	dB
I^2S IN -> DAC -> HEADPHONE OUT - 10 KOHM LOAD					
SNR (-60 dB input)		-	96	-	dB
THD+N		-	-84	-	dB
Frequency Response		-	±0.11	-	dB
PSRR (200 mVp-p @ 1.0 kHz on VDDA)		-	85	-	dB

Table 4. Audio Performance 2

Test Conditions unless otherwise noted: V_{DDIO} = 1.8 V, V_{DDA} = 1.8 V, T_A = 25°C, Slave mode, f_S = 48 kHz, MCLK = 256 f_S, 24 bit input. ADC tests were conducted with refbias = -37.5%, all other tests conducted with refbias = -50%.

Characteristic	Symbol	Min	Typ	Max	Unit
AUDIO PERFORMANCE					
Line In Input Level		-	1.0	-	V_{RMS}
Line In Input Impedance		10	-	-	kOhm
LINE IN -> ADC -> I^2S OUT					
SNR (-60 dB input)		-	90	-	dB
THD+N		-	-72	-	dB
Frequency Response		-	±0.11	-	dB
Channel Separation		-	80	-	dB
LINE IN -> HEADPHONE_LINEOUT (CODEC BYPASS MODE)					
SNR (-60 dB input)		-	102	-	dB
THD+N (10 kOhm load)		-	-89	-	dB
THD+N (16 Ohm load)		-	-87	-	dB
Frequency Response		-	±0.05	-	dB
Channel Separation (1.0 kHz)			81		dB
I^2S IN -> DAC -> LINE OUT					
Output Level		-	1.0	-	V_{RMS}
SNR (-60 dB input)		-	100	-	dB
THD+N		-	-88	-	dB
Frequency Response		-	±0.12	-	dB
I^2S IN -> DAC -> HEADPHONE OUT - 16 OHM LOAD					
Output Power		-	58	-	mW
SNR (-60 dB input)		-	98	-	dB
THD+N		-	-86	-	dB
Frequency Response		-	±0.12	-	dB
I^2S IN -> DAC -> HEADPHONE OUT - 32 OHM LOAD					
Output Power		-	30	-	mW
SNR (-60 dB input)		-	100	-	dB
THD+N		-	-88	-	dB
Frequency Response		-	±0.11	-	dB
I^2S IN -> DAC -> HEADPHONE OUT - 10 KOHM LOAD					
SNR (-60 dB input)		-	97	-	dB
THD+N		-	-85	-	dB
Frequency Response		-	±0.11	-	dB
PSRR (200 mVp-p @ 1.0 kHz on VDDA)		-	89	-	dB

DYNAMIC ELECTRICAL CHARACTERISTICS

Table 5. Dynamic Electrical Characteristics

Characteristic	Symbol	Min	Typ	Max	Unit
POWER UP TIMING					
Time from all supplies powered up and SYS_MCLK present to initial communication. See Figure 4.	t_{PC}	1.0[2]	–	-	µs
I2C BUS TIMING[3] See Figure 5.					
I^{2}C Serial Clock Frequency	f_{I2C_CLK}	-	-	400	kHz
I^{2}C Start condition hold time	t_{I2CSH}	150	-	-	ns
I^{2}C Stop condition setup time	$t_{I2CSTSU}$	150	-	-	ns
I^{2}C Data input setup time to rising edge of CTRL_CLK	t_{I2CDSU}	125	-	-	ns
I^{2}C Data input hold time from falling edge of CTRL_CLK (receiving data)	t_{I2CDH}	5.0	-	-	ns
I^{2}C Data input hold time from falling edge of CTRL_CLK (driving data)	t_{I2CDH}	360	-	-	ns
I^{2}C CTRL_CLK low time	$t_{I2CCLKL}$	300	-	-	ns
I^{2}C CTRL_CLK high time	$t_{I2CCLKH}$	100	-	-	ns
SPI BUS TIMING[4] See Figure 6.					
SPI Serial Clock Frequency	f_{SPI_CLK}	-	-	TBD	MHz
SPI data input setup time	t_{SPIDSU}	10	-	-	ns
SPI data input hold time	t_{SPIDH}	10	-	-	ns
SPI CTRL_CLK low time	$t_{SPICLKL}$	TBD	-	-	ns
SPI CTRL_CLK high time	$t_{SPICLKH}$	TBD	-	-	ns
SPI clock to chip select	t_{CCS}	60	-	-	ns
SPI chip select to clock	t_{CSC}	20	-	-	ns
SPI chip select low	t_{CSL}	20	-	-	ns
SPI chip select high	t_{CSH}	20			ns
SPECIFICATIONS AND TIMING FOR THE I^{2}S PORT[5] See Figure 7.					
Frequency of I^{2}S_LRCLK	f_{LRCLK}	TBD	-	-96	kHz
Frequency of I^{2}S_SCLK	f_{SCLK}	-	32^*f_{LRCLK} 64^*f_{LRCLK}	-	kHz
I^{2}S delay	t_{I2S_D}	-	-	10	ns
I^{2}S setup time	t_{I2S_S}	10	-	-	ns
I^{2}S hold time	t_{I2S_H}	10	-	-	ns

Notes

1. The SGTL5000 has an internal reset that is deasserted 8 SYS_MCLK cycles after all power rails have been brought up. After this time, communication can start.
2. 1.0µs represents 8 SYS_MCLK cycles at the minimum 8.0 MHz SYS_MCLK.
3. This section provides timing for the SGTL5000 while in I^{2}C mode (CTRL_MODE = 0).
4. This section provides timing for the SGTL5000 while in SPI mode (CTRL_MODE = 1)
5. The following are the specifications and timing for I^{2}S port. The timing applies to all formats.

TIMING DIAGRAMS

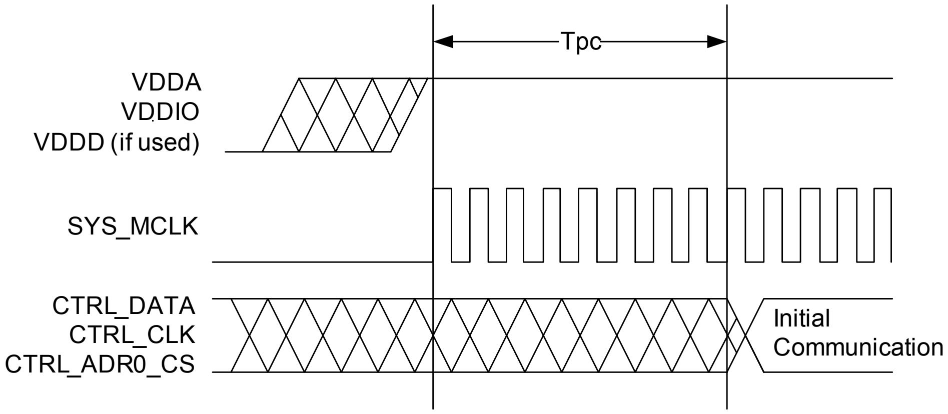


Figure 4. Power Up Timing

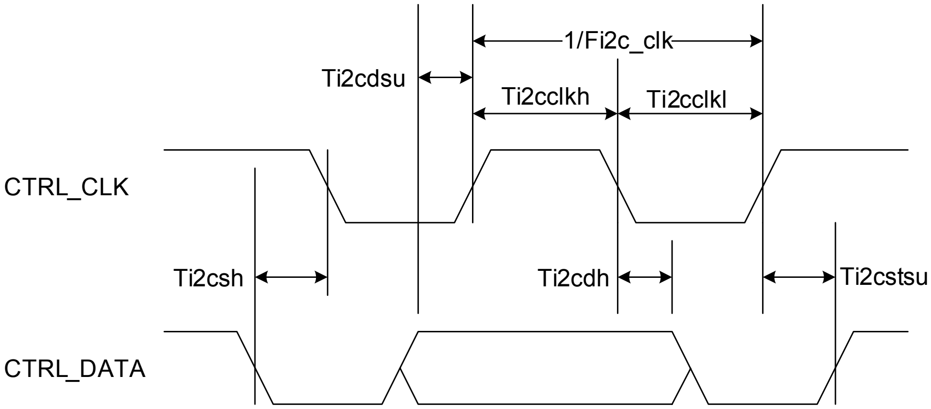


Figure 5. I^{2}C Timing (CTRL_MODE == 0)

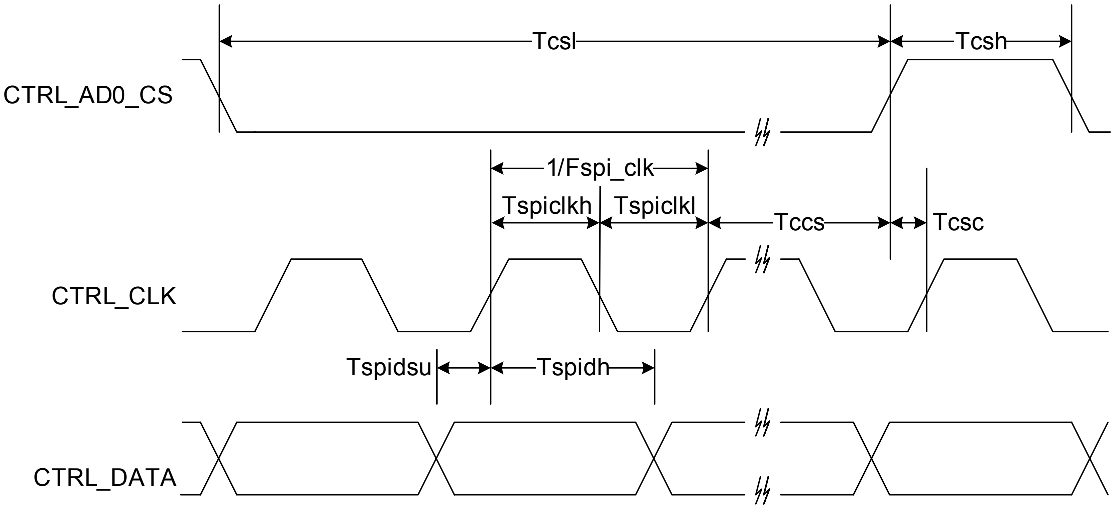


Figure 6. SPI Timing

.

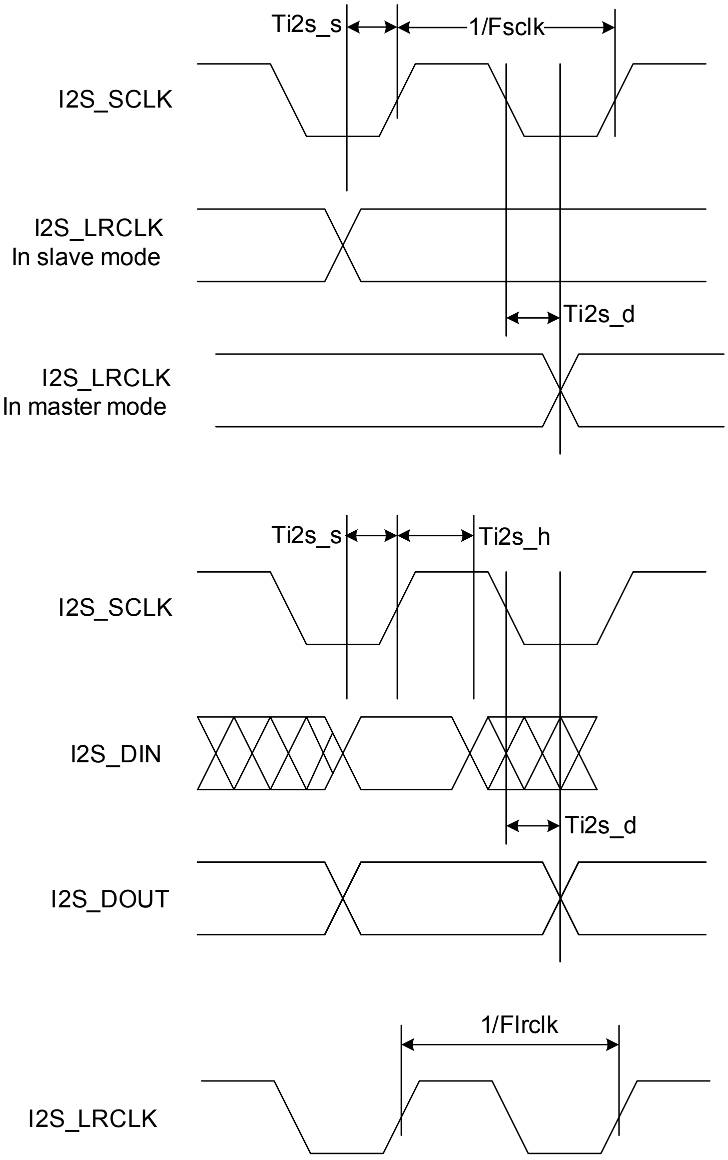


Figure 7. I^{2}S Interface Timing

FUNCTIONAL DESCRIPTION

INTRODUCTION

The SGTL5000 is a low power stereo codec with integrated headphone amplifier. It is designed to provide a complete audio solution for portable products needing line-in, mic-in, line-out, headphone-out, and digital I/O. Deriving it's architecture from best in class Freescale integrated products that are currently on the market, the SGTL5000 is able to achieve ultra low power with very high performance and functionality, all in one of the smallest footprints available. Target markets include portable media players, GPS units and smart phones. Features such as capless headphone design and USB clocking mode (12 MHz SYS_MCLK input) help lower overall system cost.

In summary, the SGTL5000 accepts the following inputs:

- Line input
- Microphone input, with mic bias (mic bias only available in 32QFN version)
- Digital I^{2}S input

In addition, the SGTL5000 supports the following outputs:

- Line output
- Headphone output
- Digital I^{2}S output

The following digital audio processing is included to allow for product differentiation:

- Digital mixer
- Freescale Surround
- Freescale Bass Enhancement
- Tone Control, parametric equalizer, and graphic equalizer

The SGTL5000 can accept an external standard master clock at a multiple of the sampling frequency (i.e. 256*Fs, 385*Fs, 512*Fs). In addition it can take non standard frequencies and use the internal PLL to derive the audio clocks. The device supports 8.0 kHz, 11.025 kHz, 16 kHz, 22.5 kHz, 24 kHz, 32 kHz, 44.1kHz, 48 kHz, 96 kHz sampling frequencies.

FUNCTIONAL INTERNAL BLOCK DESCRIPTION

SYSTEM BLOCK DIAGRAM W/ SIGNAL FLOW AND GAIN MAP

Figure 8 shows a block diagram that highlights the signal flow and gain map for the SGTL5000.

To guarantee against clipping it is important that the gain in a signal path in addition to the signal level does not exceed 0 dB at any point.

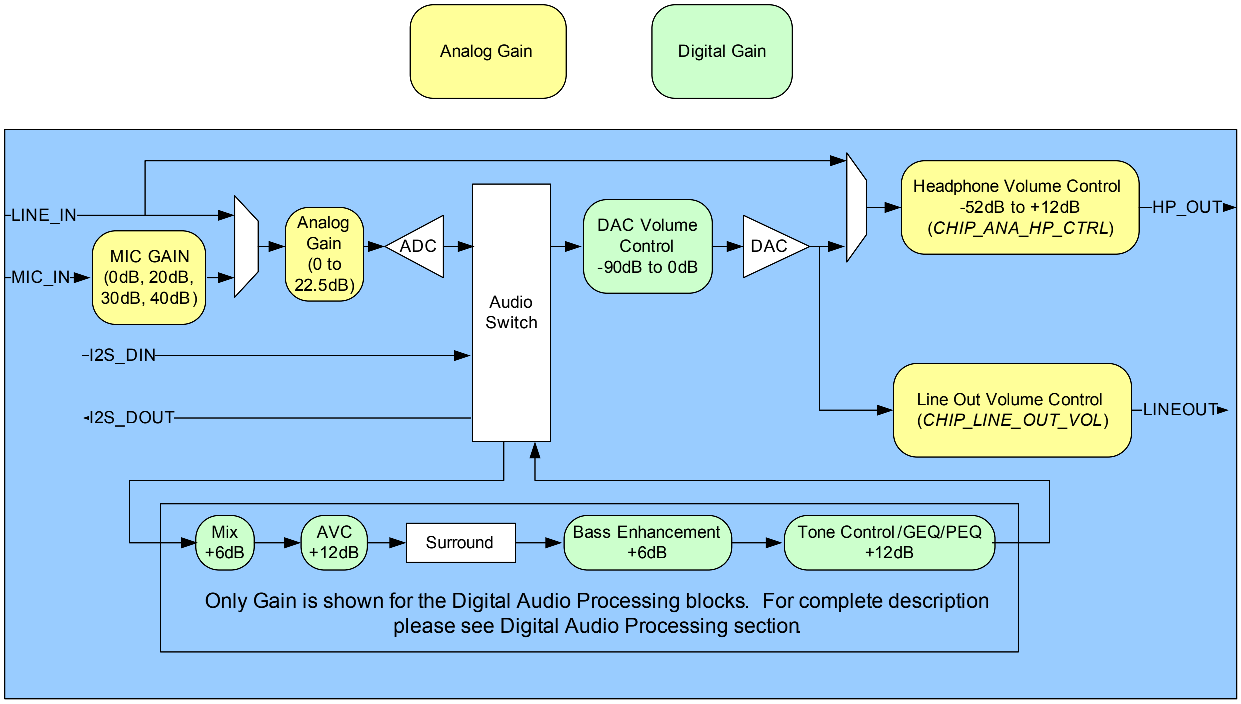

Analog Gain
Digital Gain
LINE_IN
MIC_IN
MIC GAIN (0dB, 20dB, 30dB, 40dB)
Analog Gain (0 to 22.5dB)
ADC
Audio Switch
DAC Volume Control -90dB to 0dB
DAC
Headphone Volume Control -52dB to +12dB (CHIP_ANA_HP_CTRL)
HP_OUT
I2S_DIN
I2S_DOUT
Line Out Volume Control (CHIP_LINE_OUT_VOL)
LINEOUT
Mix +6dB
AVC +12dB
Surround
Bass Enhancement +6dB
Tone Control/GEQ/PEQ +12dB
Only Gain is shown for the Digital Audio Processing blocks. For complete description please see Digital Audio Processing section

Figure 8. System Block Diagram, Signal Flow and Gain

POWER

The SGTL5000 has a flexible power architecture to allow the system designer to minimize power consumption and maximize performance at the lowest cost.

External Power Supplies

The SGTL5000 requires 2 external power supplies: VDDA and VDDIO. An optional third external power supply VDDD may be provided externally to achieve lower power. A description for the different power supplies is as follows:

- VDDA: This external power supply is used for the internal analog circuitry including ADC, DAC, LINE inputs, MIC inputs, headphone outputs and reference voltages. VDDA supply ranges are shown in Maximum Ratings. A decoupling cap should be used on VDDA, as shown in the typical application diagrams in Typical Applications.
- VDDIO: This external power supply controls the digital I/O levels as well as the output level of LINE outputs. VDDIO supply ranges are shown in Maximum Ratings. A decoupling cap should be used on VDDIO as shown in the typical application diagrams in Typical Applications.

Note that if VDDA and VDDIO are derived from the same voltage, a single decoupling capacitor can be used to minimize cost. This capacitor should be placed closest to VDDA.

- VDDD: This is a digital power supply that is used for internal digital circuitry. For a low cost design, this supply can be derived from an internal regulator and no external components are required. If no external supply is applied to VDDD, the internal regulator will automatically be used. For lowest power, this supply can be driven at the lowest specified voltage given in Maximum Ratings. If an external supply is used for VDDD, a decoupling capacitor is recommended. VDDD supply ranges are shown in Maximum Ratings, for when externally driven. If the system drives VDDD externally, an efficient switching supply should be used or no system power savings will be realized.

Internal Power Supplies

The SGTL5000 has two exposed internal power supplies, VAG and charge pump.

- VAG is the internal voltage reference for the ADC and DAC. After startup the voltage of VAG should be set to VDDA/2 by writing *CHIP_REF_CTRL->VAG_VAL*. Refer to programming Chip Powerup and Supply Configurations. The VAG pin should have an external filter capacitor as shown in the typical application diagram.
- Chargepump: This power supply is used for internal analog switches. If VDDA or VDDIO is greater than 2.7 V, this supply is automatically driven from the highest of VDDIO and VDDA. If both VDDIO and VDDA are less than 3.1 V, then the user should turn on the charge pump function to create the chargepump rail from VDDIO by writing *CHIP_ANA_POWER->VDDC_CHRGPMP_POWERUP* register. Refer to programming Chip Powerup and Supply Configurations.
- LINE_OUT_VAG is the line output voltage reference. It should be set to VDDIO/2 by writing *CHIP_LINE_OUT_CTRL->LO_VAGCNTRL*.

Power Schemes

The SGTL5000 supports a flexible architecture and allows the system designer to minimize power or maximize BOM savings.

- For maximum cost savings, all supplies can be run at the same voltage.
- Alternatively for minimum power, the analog and digital supplies can be run at minimum voltage while driving the digital I/O voltage at the voltage needed by the system.
- To save power, independent supplies are provided for line outputs and headphone outputs. This allows for 1VRMS line outputs while using minimal headphone power.
- For best power, VDDA should be run at the lowest possible voltage required for the maximum headphone output level. For highest performance, VDDA should be run at 3.3 V. For most applications a lower voltage can be used for the best performance/power combination.

RESET

The SGTL5000 has an internal reset that is deasserted 8 SYS_MCLKs after all power rails have been brought up. After this time communication can start. See Dynamic Electrical Characteristics.

CLOCKING

Clocking for the SGTL5000 is provided by a system master clock input (SYS_MCLK). SYS_MCLK should be synchronous to the sampling rate (Fs) of the I^{2}S port. Alternatively any clock between 8.0 and 27 Mhz can be provided on SYS_MCLK and the SGTL5000 can use an internal PLL to derive all internal and I^{2}S clocks. This allows the system to use an available clock such as 12 MHz (common USB clock) for SYS_MCLK to reduce overall system costs.

Synchronous SYS_MCLK input

The SGTL5000 supports various combinations of SYS_MCLK frequency and sampling frequency as shown in Table 6. Using a synchronous SYS_MCLK allows for lower power as the internal PLL is not used.

Table 6. Synchronous MCLK Rates

CLOCK	SUPPORTED RATES	UNITS
System Master Clock (SYS_MCLK)	256, 384, 512	Fs
Sampling Frequency (Fs)	8, 11.025, 16, 22.5, 32, 44.1, 48, 96 [6]	kHz

Notes

[6] For a sampling frequency of 96 kHz, only 256 Fs SYS_MCLK is supported

Using the PLL - Asynchronous SYS_MCLK input

An integrated PLL is provided in the SGTL5000 that allows any clock from 8.0 to 27 MHz to be connected to SYS_MCLK. This can help save system costs, as a clock available elsewhere in the system can be used to derive all audio clocks using the internal PLL. In this case, the clock input to SYS_MCLK can be asynchronous with the sampling frequency needed in the system. For example, a 12 MHz clock from the system processor could be used as the clock input to the SGTL5000.

Three register fields need to be configured to properly use the PLL. They are *CHIP_PLL_CTRL->INT_DIVISOR*, *CHIP_PLL_CTRL->FRAC_DIVISOR* and *CHIP_CLK_TOP_CTRL->INPUT_FREQ_DIV2*. Figure 9 shows a flowchart that shows how to determine the values to program in the register fields.

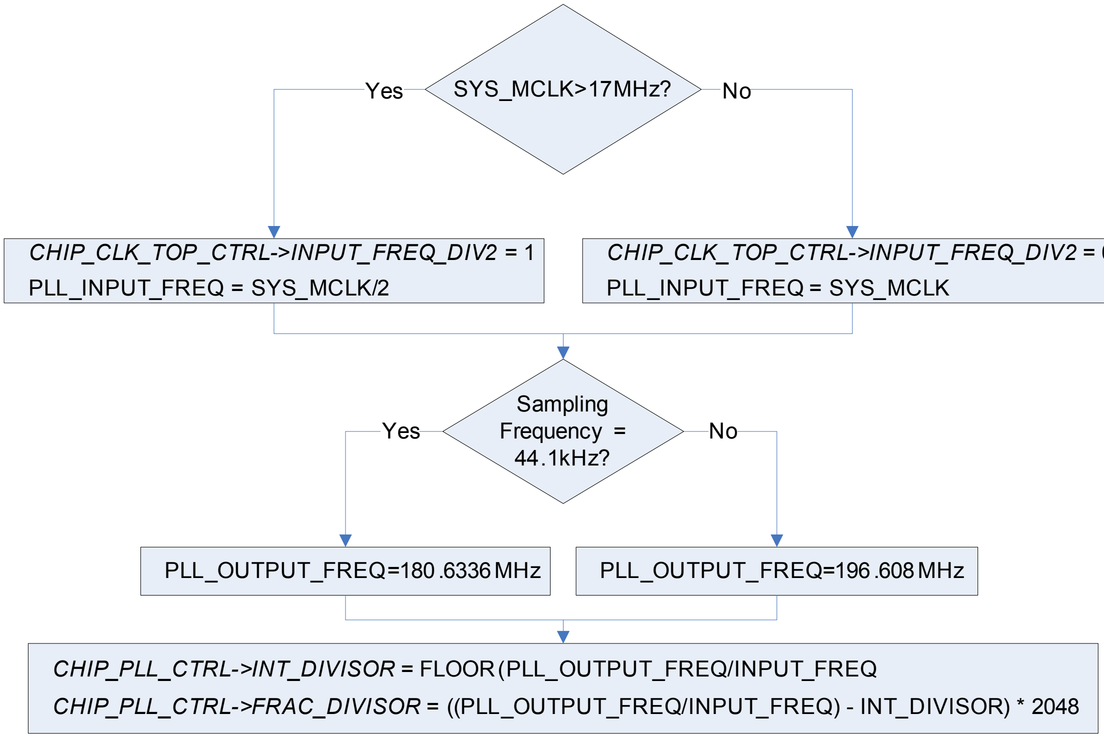

SYS_MCLK>17MHz?
Yes
No
CHIP_CLK_TOP_CTRL->INPUT_FREQ_DIV2 = 1
PLL_INPUT_FREQ = SYS_MCLK/2
CHIP_CLK_TOP_CTRL->INPUT_FREQ_DIV2 = 0
PLL_INPUT_FREQ = SYS_MCLK
Sampling Frequency = 44.1kHz?
Yes
No
PLL_OUTPUT_FREQ=180.6336MHz
PLL_OUTPUT_FREQ=196.608MHz
CHIP_PLL_CTRL->INT_DIVISOR = FLOOR(PLL_OUTPUT_FREQ/INPUT_FREQ
CHIP_PLL_CTRL->FRAC_DIVISOR = ((PLL_OUTPUT_FREQ/INPUT_FREQ) - INT_DIVISOR) * 2048

Figure 9. PLL Programming Flowchart

For example, when a 12 MHz digital signal is placed on MCLK, for a 48 kHz frame clock

CHIP_CLK_TOP_CTRL->INPUT_FREQ_DIV2 = 0 // SYS_MCLK < 17 MHz

CHIP_PLL_CTRL->INT_DIVISOR = FLOOR (196.608 MHz/12 MHz) = 16 (decimal)

CHIP_PLL_CTRL->FRAC_DIVISOR = ((196.608 MHz/ 12 MHz) - 16) * 2048 = 786 (decimal)

Refer to PLL programming PLL Configuration.

AUDIO SWITCH (SOURCE SELECT SWITCH)

The audio switch is the central routing block that controls the signal flow from input to output. Any single input can be routed to any single or multiple outputs.

Any signal can be routed to the Digital Audio Processor (DAP). The output of the DAP (an input to the audio switch) can in turn be routed to any physical output. The output of the DAP can not be routed into itself. Refer to Digital Audio Processing, for DAP information and configuration.

It should be noted that the analog bypass from Line input to headphone output does not go through the audio switch.

To configure a route, the *CHIP_SSS_CTRL* register is used. Each output from the source select switch has its own register field that is used to select what input is routed to that output.

For example, to route the I^{2}S digital input through the DAP and then out to the DAC (headphone) outputs write *SSS_CTRL->DAP_SELECT* to 0x1 (selects I2S_IN) and *SSS_CTRL->DAC_SELECT* to 0x3 (selects DAP output).

ANALOG INPUT BLOCK

The analog input block contains a stereo line input and a microphone input with mic bias (in the 32 QFN package). Either input can be routed to the ADC. The line input can also configured to bypass the CODEC and be routed the analog input directly to the headphone output.

Line Inputs

One stereo line input is provided for connection to line sources such as an FM radio or MP3 input.

The source should be connected to the left and right line inputs through series coupling capacitors. The suggested value is shown in the typical application diagram in Typical Applications.

As detailed in ADC, the line input can be routed to the ADC.

The line input can also be routed to the headphone output by writing *CHIP_ANA_CTRL->SELECT_HP*. This selection bypasses the ADC and audio switch and routes the line input directly to the headphone output to enable a very low power pass through.

Microphone Input

One mono microphone input is provided for uses such as voice recording.

Mic bias is provided in the 32QFN package. The mic bias is can be programmed with the *CHIP_MIC_CTRL->BIAS_VOLT* register field. Values from 1.25 V to 3.00 V are supported in 0.25 V steps. Mic bias should be set less than 200 mV from VDDA, e.g. with VDDA at 1.70 V, Mic bias should be set no greater than 1.50 V.

The microphone should be connected through a series coupling capacitor. The suggested value is shown in the typical connection diagram.

The microphone has programmable gain through the *CHIP_MIC_CTRL->GAIN* register field. Values of 0 dB, +20 dB, +30 dB and +40 dB are available.

ADC

The SGTL5000 contains an ADC who takes its input from either the line input or a microphone. The register field *CHIP_ANA_CTRL->SELECT_ADC* controls this selection. The output of the ADC feeds the audio switch.

The ADC has its own analog gain stage that provides 0 to +22.5 dB of gain in 1.5 dB steps. A bit is available that shifts this range down by 6.0 dB to effectively provide -6.0 dB to +16.5 dB of gain. The ADC gain is controlled in the *CHIP_ANA_ADC_CTRL* register.

The ADC has an available zero cross detect (ZCD) that will prevent any volume change until a zero-volt crossing of the audio signal is detected. This helps in eliminating pop or other audio anomalies. If the ADC is to be used, the chip reference bias current should not be set to -50% when in 3.0 V mode.

ANALOG OUTPUTS

The SGTL5000 contains a single stereo DAC that can be used to drive a headphone output and a line output. The DAC receives its input from the audio switch. The headphone output and the line output can be driven at the same time from the DAC.

The headphone output can also be driven directly by the line input bypassing the ADC and DAC for a very low power mode of operation.

The headphone output is powered by VDDA while the line output is powered by VDDIO. This allows the headphone output to be run at the lowest possible voltage while the line output can still meet line output level requirements.

DAC

The DAC output is routed to the headphone and the dedicated line output.

The DAC output has a digital volume control from -90 dB to 0 dB in ~0.5 dB step sizes. This volume is shared among headphone output and line output. The register *CHIP_DAC_VOL* controls the DAC volume.

Headphone

Stereo headphone outputs are provided which can be used to drive a headphone load or a line level output. The headphone output has its own independent analog volume control with a volume range of -52 dB to +12 dB in 0.5 dB step sizes. This volume control can be used in addition to the DAC volume control. For best performance the DAC volume control should be left at 0 dB until the headphone is brought to its lowest setting of -52 dB. The register *CHIP_ANA_HP_CTRL* is used to control the headphone volume.

The headphone output has an independent mute that is controlled by the register field *CHIP_ANA_CTRL->MUTE_HP*.

The line input is routed to the headphone output by writing *CHIP_ANA_CTRL->SELECT_HP*. This selection bypasses the ADC and audio switch and routes the line input directly to the headphone output to enable a very low power pass through. When the line input is routed to the headphone output, only the headphone analog volume and mute will affect the headphone output.

The headphone has an available zero cross detect (ZCD) which, as previously described, will prevent any volume change until a zero-volt crossing of the audio signal is detected. This helps in eliminating pop or other audio anomalies.

Line Outputs

The SGTL5000 contains a stereo line output. The line output has a dedicated gain stage that can be used to adjust the output level. The *CHIP_LINE_OUT_VOL* controls the line level output gain.

The line outputs also have a dedicated mute that is controlled by the register field *CHIP_ANA_CTRL->MUTE_LO.*

The line out volume is intended as maximum output level adjustment. It is intended to be used to set the maximum output swing. It does not have the range of a typical volume control and does not have a zero cross detect (ZCD). However the dac digital volume could be used if volume control is desired.

FUNCTIONAL DEVICE OPERATION

POWER CONSUMPTION

Table 7. Power Consumption: V_{DDA}=1.8 V, V_{DDIO}=1.8 V

MODE	CURRENT CONSUMPTION (MA)			POWER (MW)
	V_{DDD}	V_{DDA}	V_{DDIO}	
Playback (I^2S->DAC->Headphone)	-	2.54	0.9	6.19
Playback with DAP ((I^2S->DAP->DAC->Headphone)	-	3.59	0.9	8.08
Playback/Record (I^2S->DAC->Headphone, ADC->I^2S)	-	3.71	1.10	8.67
Record (ADC->I^2S)	-	2.29	1.06	6.02
Analog playback, CODEC bypassed (LINEIN->HP)	-	1.48	0.89	4.27
Standby, all analog power off	-	0.019	0.002	0.038
Playback with PLL (I^2S->DAC->HP)	-	3.01	2.17	9.31

V_{DDD} derived internally @ 1.2 V, slave mode except for PLL case, 32 ohm load on HP, Conditions: -100 dBFs signal input, slave mode unless otherwise noted, paths tested as indicated, unused paths turned off.

A further 0.5-1.0 mW reduction in power is expected with TA2 silicon.

Table 8. Power Consumption: V_{DDA}=3.3 V, V_{DDIO}=3.3 V

MODE	CURRENT CONSUMPTION (MA)			POWER(MW)
	V_{DDD}	V_{DDA}	V_{DDIO}	
Playback (I^2S->DAC->Headphone)	-	3.45	0.067	11.60
Playback with DAP ((I^2S->DAP->DAC->Headphone)	-	4.49	0.067	15.03
Playback/Record (I^2S->DAC->Headphone, ADC->I^2S)	-	4.67	0.343	16.53
Record (ADC->I^2S)	-	2.90	0.296	10.56
Analog playback, CODEC bypassed (LINEIN->HP)	-	1.91	0.039	6.43
Standby, all analog power off	-	0.04	0.002	0.139
Playback with PLL (I^2S->DAC->HP)	-	3.92	2.76	22.05

DIGITAL INPUT & OUTPUT

One I^2S (Digital Audio) Port is provided which supports the following formats: I^2S, Left Justified, Right Justified, and PCM mode.

I^2S, Left Justified, and Right Justified Modes

I^2S, Left Justified and Right Justified modes are stereo interface formats. The I2S_SCLK frequency, I2S_SCLK polarity, I2S_DIN/DOUT data length, and I2S_LRCLK polarity can all be change through the *CHIP_I2S_CTRL* register. For I2S, Left Justified and Right Justified formats the left subframe should always be presented first regardless of the *CHIP_I2S_CTRL->LRPOL* setting.

The I2S_LRCLK and I2S_SCLK can be programmed as master (driven to an external target) or slave (driven from an external source). When the clocks are in slave mode, they must be synchronous to SYS_MCLK. For this reason the SGTL5000 can only operate in synchronous mode (see Clocking) while in I^{2}S slave mode.

In master mode, the clocks will be synchronous to SYS_MCLK or the output of the PLL when the part is running in asynchronous mode.

Figure 10 shows functional examples of different common digital interface formats and their associated register settings.

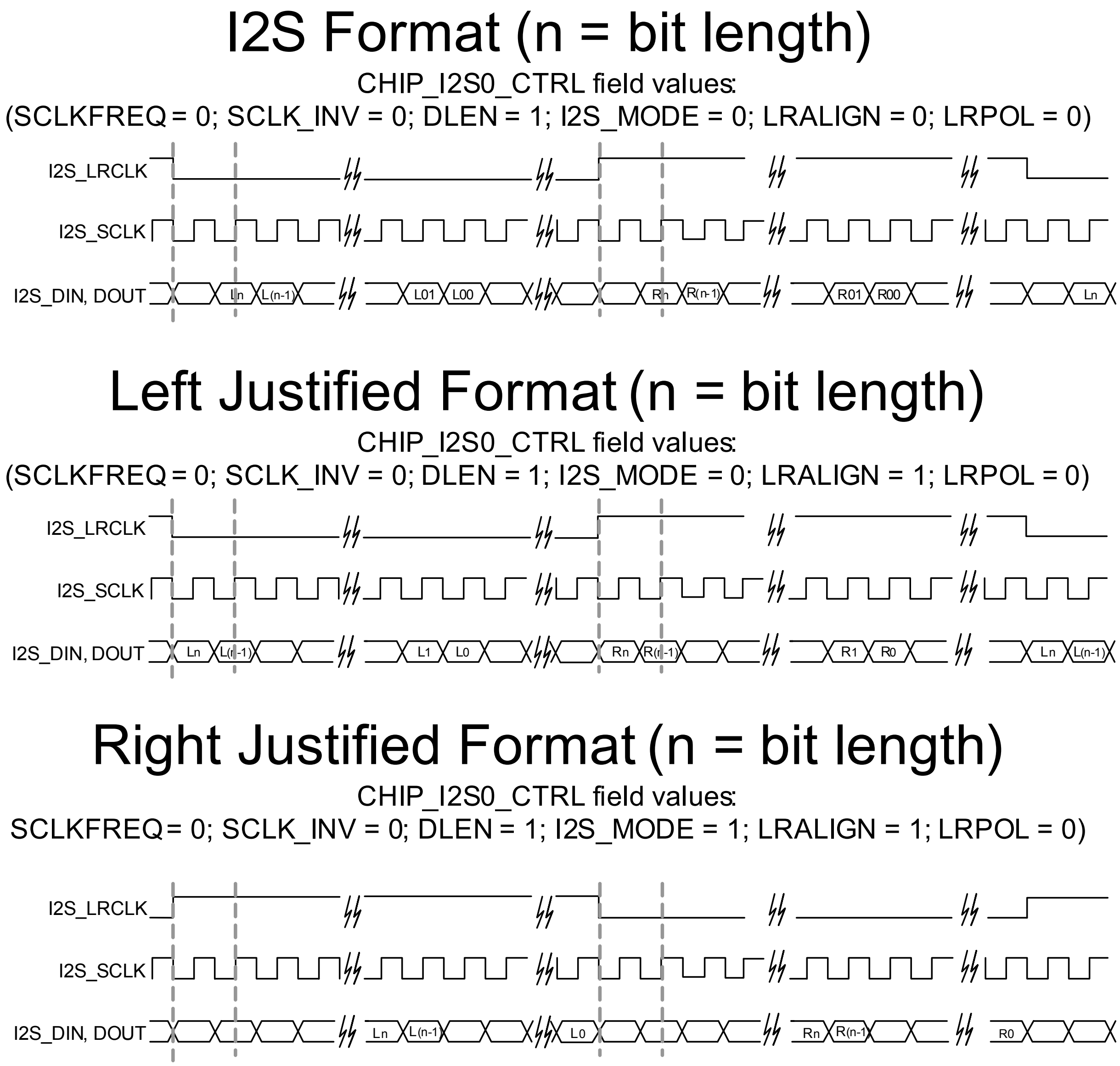

I2S Format (n = bit length)
CHIP_I2S0_CTRL field values:
(SCLKFREQ = 0; SCLK_INV = 0; DLEN = 1; I2S_MODE = 0; LRALIGN = 0; LRPOL = 0)
I2S_LRCLK
I2S_SCLK
I2S_DIN, DOUT
Ln
L(n-1)
L01
L00
Rn
R(n-1)
R01
R00
Ln
Left Justified Format (n = bit length)
CHIP_I2S0_CTRL field values:
(SCLKFREQ = 0; SCLK_INV = 0; DLEN = 1; I2S_MODE = 0; LRALIGN = 1; LRPOL = 0)
I2S_LRCLK
I2S_SCLK
I2S_DIN, DOUT
Ln
L(n-1)
L1
L0
Rn
R(n-1)
R1
R0
Ln
L(n-1)
Right Justified Format (n = bit length)
CHIP_I2S0_CTRL field values:
SCLKFREQ = 0; SCLK_INV = 0; DLEN = 1; I2S_MODE = 1; LRALIGN = 1; LRPOL = 0)
I2S_LRCLK
I2S_SCLK
I2S_DIN, DOUT
Ln
L(n-1)
L0
Rn
R(n-1)
R0

Figure 10. I^{2}S Port Supported Formats

PCM Mode

The I^{2}S port can also be configured into a PCM mode (also known as DSP mode). This mode is provided to allow connectivity to external devices such as Bluetooth modules. PCM mode differs from other interface formats presented in I2S, Left Justified, and Right Justified Modes, in that the frame clock (I2S_LRCLK) does not represent a different channel when high or low, but is a bit-wide pulse that marks the start of a frame. Data is aligned such that the left channel data is immediately followed by right channel data. Zero padding is filled in for the remaining bits. The data and frame clock may be configured to clock in on the rising or falling edge of Bit Clock.

PCM Format A signifies the data word beginning one SCLK bit following the I2S_LRCLK transition, as in I^{2}S Mode. PCM Format B signifies the data word beginning after the I2S_LRCLK transition, as in Left Justified.

In slave mode, the pulse width of the I2S_LRCLK does not matter. The pulse can range from one cycle high to all but one cycle high. In master mode, it will be driven one cycle high.

Figures 11 shows a functional drawing of the different formats in master mode.

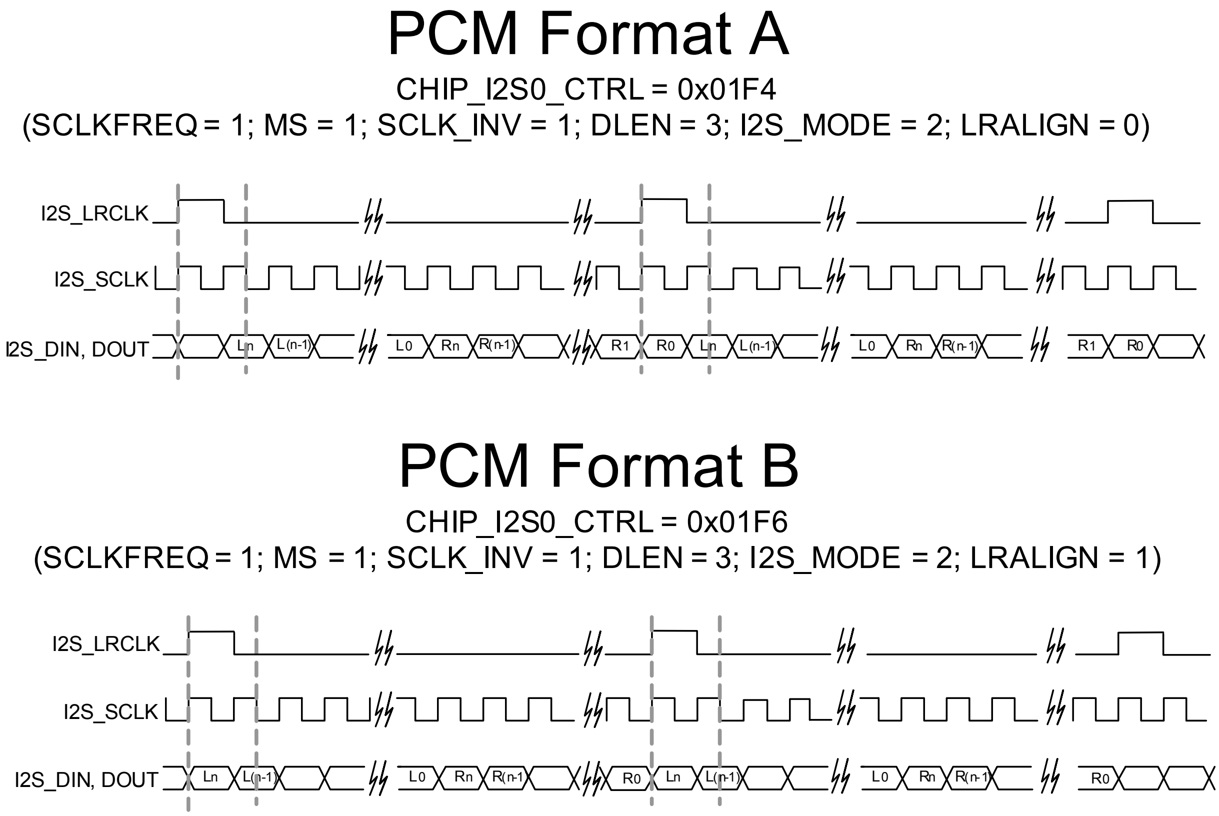


Figure 11. PCM Formats

DIGITAL AUDIO PROCESSING

The SGTL5000 contains a digital audio processing block (DAP) attached to the source select switch. The digitized signal from the source select switch can be routed into the DAP block for audio processing. The DAP has the following 5 sub blocks:

- Dual Input Mixer
- Freescale Surround
- Freescale Bass Enhancement
- 7-Band Parameter EQ / 5-Band Graphic EQ / Tone Control (only one can be used at a time)
- Automatic Volume Control (AVC)

The block diagram in Figure 12 shows the sequence in which the signal passes through these blocks.

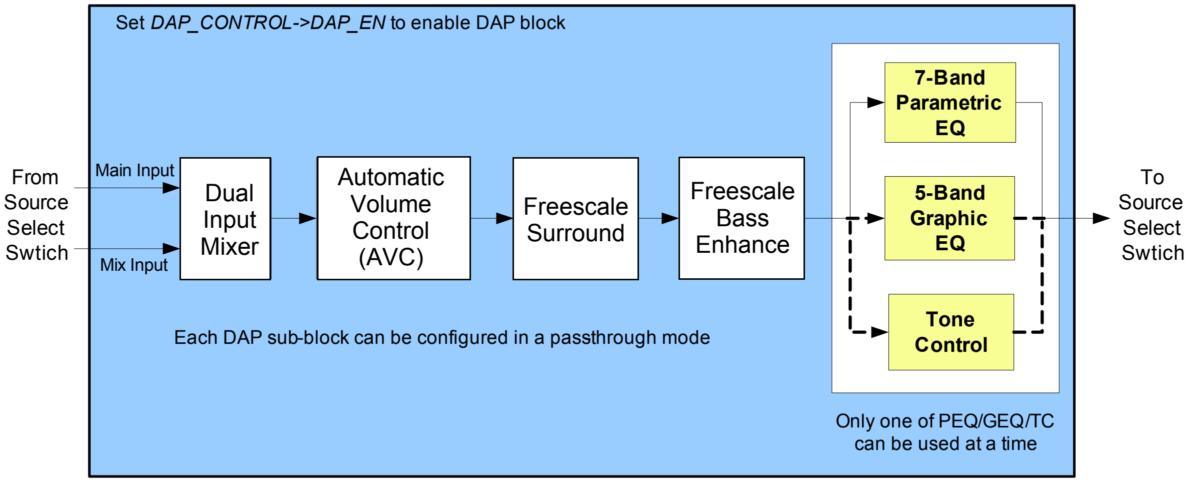

Set *DAP_CONTROL->DAP_EN* to enable DAP block
From Source Select Swtich
Main Input
Mix Input
Dual Input Mixer
Automatic Volume Control (AVC)
Freescale Surround
Freescale Bass Enhance
7-Band Parametric EQ
5-Band Graphic EQ
Tone Control
To Source Select Swtich
Each DAP sub-block can be configured in a passthrough mode
Only one of PEQ/GEQ/TC can be used at a time

Figure 12. Digital Audio Processing Block Diagram

When the DAP block is added in the route, it must be enabled separately to get audio through. It is recommended to mute the outputs before enabling/disabling the DAP block to avoid any pops or clicks due to discontinuities in the output.

Refer to Digital Audio Processor Configuration for programming examples on how to enable/disable the DAP block.

Each sub-block of the DAP can be individually disabled if its processing is not required. The sections below describes the DAP sub-blocks and how to configure them.

Dual Input Mixer

The dual input digital mixer allows for two incoming streams from the source select switch as shown in DAP - Dual Input Mixer.

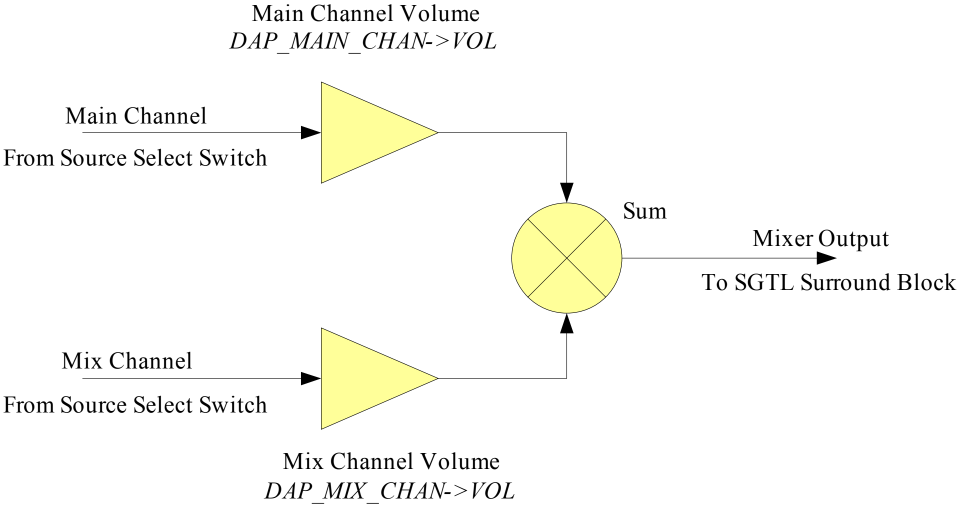


Figure 13. DAP - Dual Input Mixer

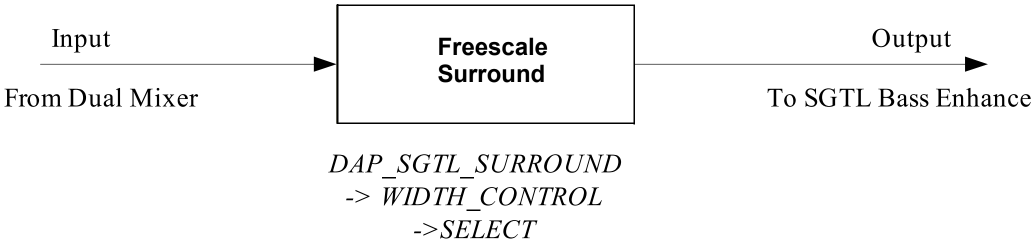
The Dual Input Mixer can be enabled or configured in a pass-through mode (Main channel will be passed through without any mixing). When enabled, the volume of the main and mix channels can be independently controlled before they are mixed together.

The volume range allowed on each channel is 0% to 200% of the incoming signal level. The default is 100% (same as input signal level) volume on the main input and 0% (muted) on the mix input.

Refer to Dual Input Mixer for programming examples on how to enable/disable the mixer and also to set the main and mix channel volume.

Freescale Surround

Freescale Surround is a royalty free virtual surround algorithm for stereo or mono inputs. It widens and deepens sound stage for music input.



The SGTL Surround can be enabled or configured in pass-through mode (input will be passed through without any processing). When enabling the Surround, mono or stereo input type must be selected based on the input signal. Surround width may be adjusted for the size of the sound stage.

Refer to Freescale Surround and Freescale Surround On/Off for a programming example on how to configure Surround width and how to enable/disable Surround.

Freescale Bass Enhance

Freescale Bass Enhance is a royalty-free algorithm that enhances natural bass response of the audio. Bass Enhance extracts bass content from right and left channels, adds bass and mixes this back up with the original signal. An optional complementary high pass filter is provided after the mixer.

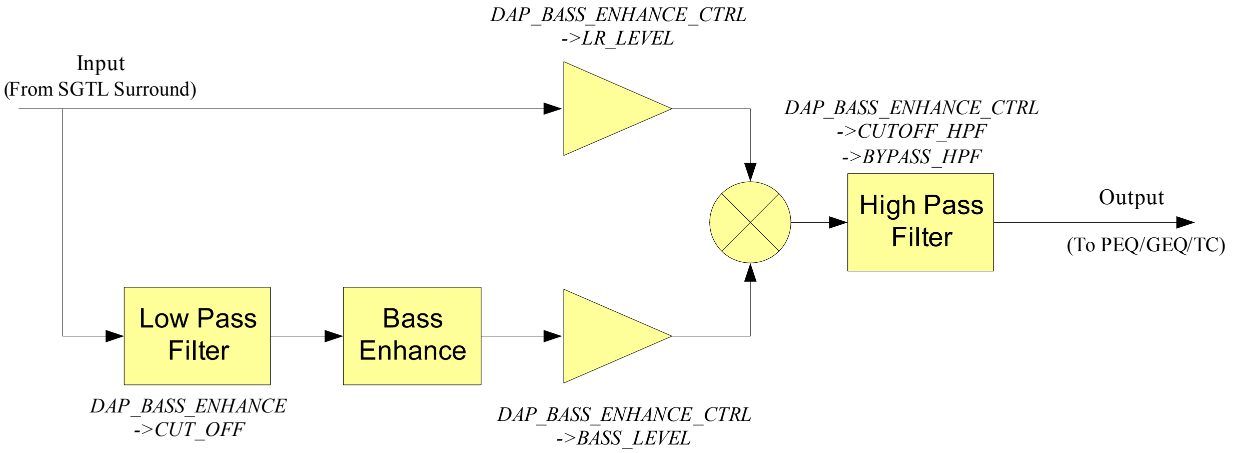


Figure 14. DAP- Freescale Bass Enhance

The SGTL Bass Enhance can be enabled or configured in pass-through mode (input will be passed through without any processing).

The cutoff frequency of the low-pass filter (LPF) can be selected based on the speakers frequency response. The cutoff frequency of the low-pass and high-pass filters are selectable between 80 to 225 Hz. Also, the input signal and bass enhanced signal can be individually adjusted for level before the two signals are mixed.

Refer to Freescale Bass Enhance and Bass Enhance On/Off for a programming example on how to configure Bass Enhance and how to enable/disable this feature.

7-Band Parametric EQ / 5-Band Graphic EQ / Tone Control

One 7-band parametric equalizer (PEQ) and one 5-band graphic equalizer (GEQ) and a Tone Control (Bass and Treble control) blocks are implemented as mutually exclusive blocks. Only one block can be used at a given time.

Refer to 7-Band Parametric EQ / 5-Band Graphic EQ / Tone Control for a programming example that shows how to select the desired EQ mode.

7-Band Parametric EQ

The 7-band PEQ allows the designer to compensate for speaker response and to provide the ability to filter out resonant frequencies caused by the physical system design. The system designer can create custom EQ presets such Rock, Speech, Classical, etc, that allows the users the flexibility in customizing their audio.

The 7-band PEQ is implemented using 7 cascaded second order IIR filters.   All filters are implemented using programmable bi-quad filters. Figure 15 shows the transfer function and Direct Form 1 of the five coefficient biquadratic filter.

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

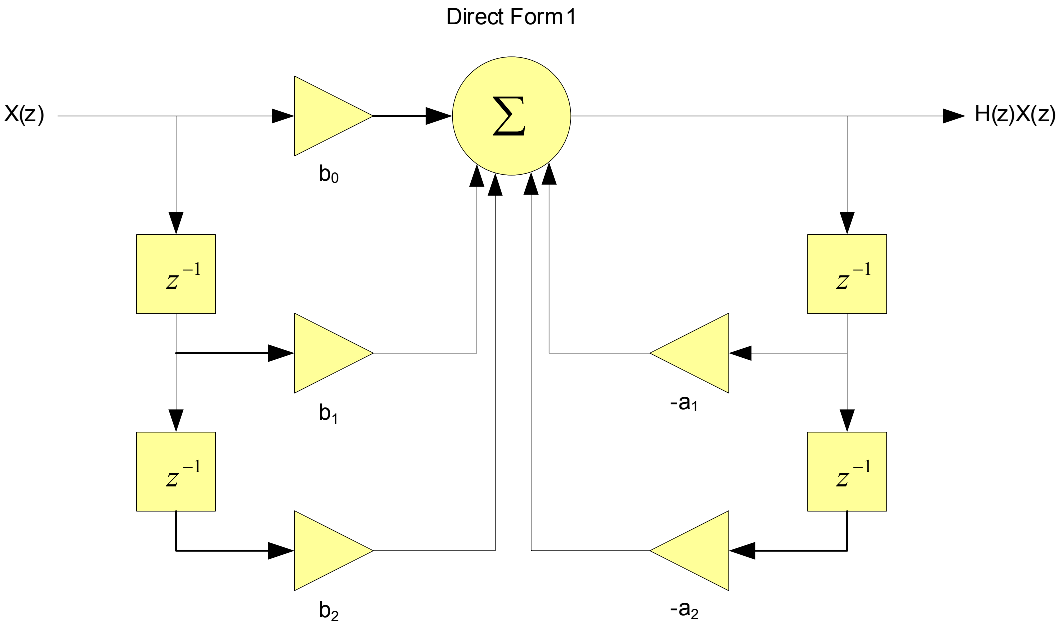


Figure 15. 5-Coefficient Biquad Filter and Transfer Function

If a band is enabled but is not being used (flat response), then a value of 0.5 should be put in b_0 and all other coefficients should be set to 0.0. Note that the coefficients must be converted to hex values before writing to the registers. By default, all the filters are loaded with coefficients to give a flat response.

In order to create EQ presets such as Rock, Speech, Classical, etc, the coefficients must be calculated, converted to 20-bit hex values and written to the registers. Note that coefficients are sample-rate dependent and separate coefficients must be generated for different sample rates. Please contact Freescale for assistance with generating the coefficients.

Refer to 7-Band PEQ Preset Selection for a programming example that shows how load the filter coefficients when the end-user changes the preset.

PEQ can be disabled (pass-through mode) by writing 0 to DAP_AUDIO_EQ->EN bits.

5-Band Graphic EQ

The 5-band graphic equalizer is implemented using 5 parallel second order IIR filters. All filters are implemented using biquad filters whose coefficients are programmed to set the bands at specific frequency. The GEQ bands are fixed at 115 Hz, 330 Hz, 990 Hz, 3000 Hz, and 9900 Hz. The volume on each band is independently adjustable in the range of +12 dB to -11.75 dB in 0.25 dB steps.

Refer to 5-Band GEQ Volume Change for a programming example that shows how to change the GEQ volume.

Tone Control

Tone control comprises treble and bass controls. The tone control is implemented as one 2nd order low pass filter (bass) and one 2nd order high pass filter (treble).

Refer to Tone Control - Bass and Treble Change for a programming example that shows how to change Bass and Treble values.

Automatic Volume Control (AVC)

An Automatic Volume Control (AVC) block is provided to reduce loud signals and amplify low level signals for easier listening. The AVC is designed to compress audio when the measured level is above the programmed threshold or to expand the audio to the programmed threshold when the measured audio is below the threshold. The threshold level is programmable with allowed range of 0 to -96 dB. Figure 16 shows the AVC block diagram and controls.

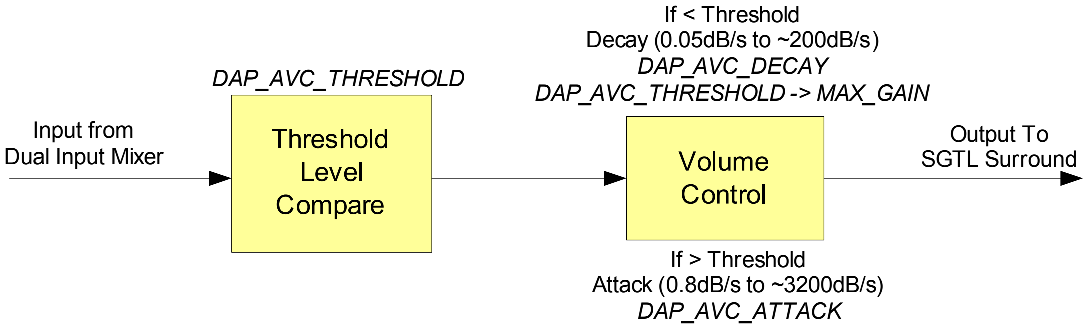


Figure 16. DAP AVC Block Diagram

When the measured audio level is below threshold, the AVC can apply a maximum gain of up to 12 dB. The maximum gain can be selected, either 0, 6, or 12 dB. When the maximum gain is set to 0 dB the AVC acts as a limiter. In this case the AVC will only take effect when the signal level is above the threshold.

The rate at which the incoming signal is attenuated down to the threshold is called the attack rate. Too high of an attack will cause an unnatural sound as the input signal is distorted. Too low of an attack may cause saturation of the output as the incoming signal will not be compressed quickly enough. The attack rate is programmable with allowed range of 0.05 dB/s to 200 dB/s.

When the signal is below the threshold, AVC will adjust the volume up until either the threshold or the maximum gain is reached. The rate at which this volume is changed is called the decay rate. The decay rate is programmable with allowed range of 0.8 dB/s to 3200 dB/s. It is desirable to use very slow decay rate to avoid any distortion in the signal and prevent the AVC from entering a continuous attack-decay loop.

Refer to Automatic Volume Control (AVC) and Automatic Volume Control (AVC) On/Off for a programming example that shows how to configure AVC and how to enable/disable AVC respectively.

CONTROL

The SGTL5000 supports both I^{2}C and SPI control modes. The CTRL_MODE pin chooses which mode will be used. When CTRL_MODE is tied to ground, the control mode is I^{2}C. When CTRL_MODE is tied to VDDIO, the control mode is SPI.

Regardless of the mode, the control interface is used for all communication with the SGTL5000 including startup configuration, routing, volume, etc.

I^{2}C

The I^{2}C port is implemented according to the I^{2}C specification v2.0. The I^{2}C interface is used to read and write all registers.

For the 32 QFN version of the SGTL5000, the I^{2}C device address is 0n01010(R/W) where n is determined by I2C_ADR0_CS and R/W is the read/write bit from the I^{2}C protocol.

For the 20 QFN version of the SGTL5000 the I^{2}C address is always 0001010(R/W).

The SGTL5000 is always the slave on all transactions which means that an external master will always drive CTRL_CLK.

In general an I^{2}C transaction looks as follows.

All locations are accessed with a 16 bit address. Each location is 16 bits wide.

An example I^{2}C write transaction follows:

- Start condition
- Device address with the R/W bit cleared to indicate write
- Send two bytes for the 16 bit register address (most significant byte first)
- Send two bytes for the 16 bits of data to be written to the register (most significant byte first)
- Stop condition

An I^{2}C read transaction is defined as follows:

- Start condition
- Device address with the R/W bit cleared to indicate write
- Send two bytes for the 16 bit register address (most significant byte first)
- Stop Condition followed by start condition (or a single restart condition)
- Device address with the R/W bit set to indicate read
- Read two bytes from the addressed register (most significant byte first)
- Stop condition

Figure 17 shows the functional I^{2}C timing diagram.

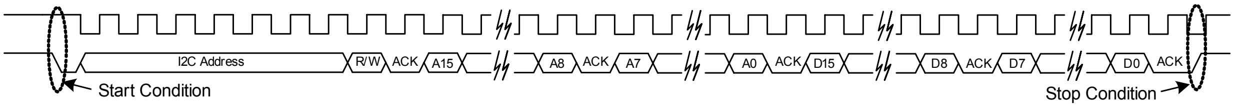


Figure 17. Functional I^{2}C Diagram

The protocol has an auto increment feature. Instead of sending the stop condition after two bytes of data, the master may continue to send data byte pairs for writing, or it may send extra clocks for reading data byte pairs. In either case, the access address is incremented after every two bytes of data. A start or stop condition from the I^{2}C master interrupts the current command. For reads, unless a new address is written, a new start condition with R/W=0 reads from the current address and continues to auto increment.

The following diagrams describe the different access formats. The gray fields are from the I^{2}C master, and the white fields are the SGTL5000 responses. Data [n] corresponds to the data read from the address sent, data[n+1] is the data from the next register, and so on.

S = Start Condition

Sr = Restart Condition

A = Ack

N = Nack

P = Stop Condition

TA2 silicon will allow for up to a 3.6 V I^{2}C signal level, regardless of the VDDIO level.

Table 9. Write Single Location

S	Device Address	W (0)	A	ADDR byte 1	A	ADDR byte 0	A	DATA byte 1	A	DATA byte 0	A	P

Table 10. Write Auto increment

S	Device Address	W (0)	A	start ADDR byte 1	A	start ADDR byte 0	A	DATA [n] byte 1	A	DATA [n] byte 0	A	DATA [n+1] byte 1	A	DATA [n+1] byte 0	A	P

Table 11. Read Single Location

S	Device Address	W (0)	A	ADDR byte 1	A	ADDR byte 0	A	Sr	Device Address	R (1)	A	DATA byte 1	A	DATA byte 0	N	P

Table 12. Read Auto increment

S	Device Address	W (0)	A	start ADDR byte 1	A	start ADDR byte 0	A	Sr	Device Address	R (1)	A	DATA [n] byte 1	A	DATA [n] byte 0	A	DATA [n+1] byte 1	A	DATA [n+1] byte 0	N	P

Table 13. Read Continuing Auto increment

S	Device Address	R	A	DATA [n+2] byte 1	A	DATA [n+2] byte 0	A	DATA [n+3] byte 1	A	DATA [n+3] byte 0	N	P

SPI

Serial Peripheral Interface (SPI) is a communications protocol supported by the SGTL5000. The SGTL5000 is always a slave. The CTRL_AD0_CS is used as the slave select (SS) when the master wants to select the SGTL5000 for communication. CTRL_CLK is connected to master's SCLK and CTRL_DATA is connected to master's MOSI line.

The part only supports allows SPI write operations and does not support read operations.

Figure 18 shows the functional timing diagram of the SPI communication protocol as supported by SGTL5000 chip. Note that on the rising edge of the SS, the chip latches to previous 32 bits of data. It interprets the latest 16-bits as register value and 16-bits preceding it as register address.

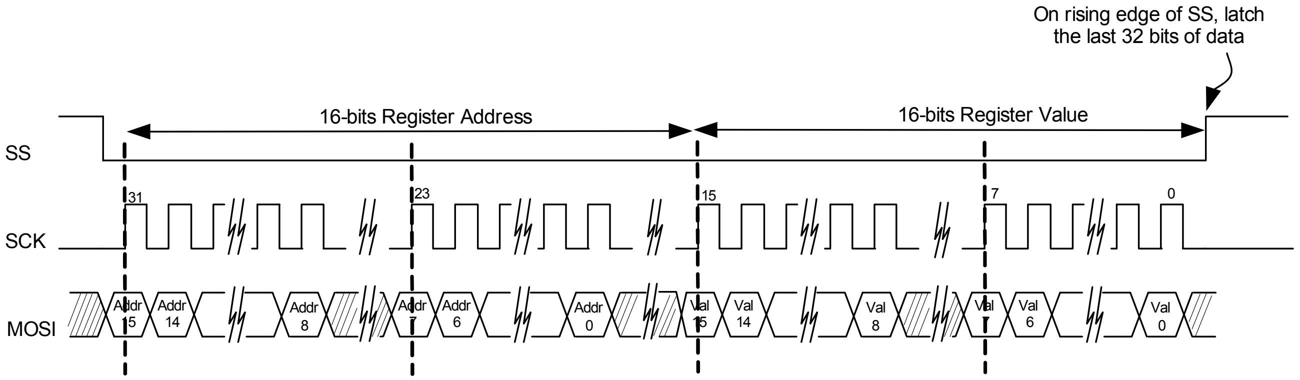


Figure 18. Functional Timing Diagram of SPI Protocol

PROGRAMMING EXAMPLES

This section provides programming examples showing how to configure the chip. The registers can be written/read by using I^{2}C communication protocol. The chip also supports SPI communication protocol, but only register write operation is supported.

PROTOTYPE FOR READING AND WRITING A REGISTER

The generic register read write prototype will be used throughout this section as shown below. The I^{2}C or SPI implementation will be specific to the I^{2}C/SPI hardware used in the system.

```
// This prototype writes a value to the entire register. All
// bit-fields of the register will be written.
Write REGISTER REGISTERVALUE
// This prototype writes a value only to the bit-field specified.
// In the actual implementation, the other bit-fields should be
// masked to prevent them from being written. Also, the
// actual implementation should left-shift the BITFIELDVALUE
// by appropriate number to match the starting bit location of
// the BITFIELD.
Modify REGISTER -> BITFIELD,   BITFIELDVALUE //Bitfield
Location
// Example implementation
// Modify DAP_EN (bit 0) bit to value 1 to enable DAP block
Modify(DAP_CONTROL_REG, 0xFFFE, 1 <<
DAP_EN_STARTBIT);
// Example Implementation of Modify
void Modify(unsigned short usRegister,
            unsigned short usClearMask,
            unsigned short usSetValue)
{
 unsigned short usData;
 // 1) Read current value
 ReadRegister(usRegister, &usData);
 // 2) Clear out old bits
 usData = usData & usClearMask;
 // 3) set new bit values
 usData = usData | usSetValue;
 // 4) Write out new value created
 WriteRegister(usRegister, usData);
}
```

CHIP CONFIGURATION

All outputs (LINEOUT, HP_OUT, I2S_OUT) are muted by default on power up. To avoid any pops/clicks, the outputs should remain muted during these chip configuration steps. Volume Control for volume and mute control.

Initialization

Chip Powerup and Supply Configurations

After the power supplies for chip is turned on, following initialization sequence should be followed. Please note that certain steps may be optional or different values may need to be written based on the power supply voltage used and desired configuration. The initialization sequence below assumes VDDIO = 3.3 V and VDDA = 1.8 V.

```
//--------------- Power Supply Configuration----------------
// NOTE: This next 2 Write calls is needed ONLY if VDDD is
// internally driven by the chip
// Configure VDDD level to 1.2V (bits 3:0)
Write CHIP_LINREG_CTRL   0x0008
// Power up internal linear regulator (Set bit 9)
Write CHIP_ANA_POWER     0x7260
// NOTE: This next Write call is needed ONLY if VDDD is
// externally driven
// Turn off startup power supplies to save power (Clear bit 12 and
13)
Write CHIP_ANA_POWER     0x4260
// NOTE: The next 2 Write calls is needed only if both VDDA and
// VDDIO power supplies are less than 3.1V.
// Enable the internal oscillator for the charge pump (Set bit 11)
Write CHIP_CLK_TOP_CTRL   0x0800
// Enable charge pump (Set bit 11)
Write CHIP_ANA_POWER     0x4A60
// NOTE: The next 2 modify calls is only needed if both VDDA and
// VDDIO are greater than 3.1 V
// Configure the chargepump to use the VDDIO rail (set bit 5 and
bit 6)
Write CHIP_LINREG_CTRL   0x006C
//---- Reference Voltage and Bias Current Configuration----
// NOTE: The value written in the next 2 Write calls is dependent
// on the VDDA voltage value.
// Set ground, ADC, DAC reference voltage (bits 8:4). The value
should
// be set to VDDA/2.   This example assumes VDDA = 1.8 V.
VDDA/2 = 0.9 V.
// The bias current should be set to 50% of the nominal value (bits
3:1)
Write CHIP_REF_CTRL 0x004E
// Set LINEOUT reference voltage to VDDIO/2 (1.65 V) (bits 5:0)
and bias current (bits 11:8) to the recommended value of 0.36 mA
for 10 kOhm load with 1.0 nF capacitance
Write CHIP_LINE_OUT_CTRL 0x0322
//------------Other Analog Block Configurations--------------
// Configure slow ramp up rate to minimize pop (bit 0)
Write CHIP_REF_CTRL      0x004F
// Enable short detect mode for headphone left/right
// and center channel and set short detect current trip level
// to 75 mA
Write CHIP_SHORT_CTRL     0x1106
// Enable Zero-cross detect if needed for HP_OUT (bit 5) and ADC
(bit 1)
```

```
Write CHIP_ANA_CTRL      0x0133
//------------Power up Inputs/Outputs/Digital Blocks---------
// Power up LINEOUT, HP, ADC, DAC
Write CHIP_ANA_POWER 0x6AFF
// Power up desired digital blocks
// I2S_IN (bit 0), I2S_OUT (bit 1), DAP (bit 4), DAC (bit 5),
// ADC (bit 6) are powered on
Write CHIP_DIG_POWER      0x0073
//----------------Set LINEOUT Volume Level-------------------
// Set the LINEOUT volume level based on voltage reference (VAG)
// values using this formula
// Value = (int)(40*log(VAG_VAL/LO_VAGCNTRL) + 15)
// Assuming VAG_VAL and LO_VAGCNTRL is set to 0.9 V and 1.65 V respectively, the // left LO vol (bits 12:8) and right LO volume (bits 4:0) value should be set // to 5
Write CHIP_LINE_OUT_VOL 0x0505
```

System MCLK and Sample Clock

```
// Configure SYS_FS clock to 48 kHz
// Configure MCLK_FREQ to 256*Fs
Modify CHIP_CLK_CTRL->SYS_FS 0x0002 // bits 3:2
Modify CHIP_CLK_CTRL->MCLK_FREQ 0x0000 // bits 1:0
// Configure the I$^2$S clocks in master mode
// NOTE: I$^2$S LRCLK is same as the system sample clock
Modify CHIP_I2S_CTRL->MS 0x0001 // bit 7
```

PLL Configuration

These programming steps are needed only when the PLL is used. Using the PLL - Asynchronous SYS_MCLK input for details on when to use the PLL.

To avoid any pops/clicks, the outputs should be muted during these chip configuration steps. Refer to Volume Control for volume and mute control.

```
// Power up the PLL
Modify CHIP_ANA_POWER->PLL_POWERUP 0x0001 // bit 10
Modify CHIP_ANA_POWER->VCOAMP_POWERUP 0x0001 // bit 8
// NOTE: This step is required only when the external SYS_MCLK
// is above 17 MHz. In this case the external SYS_MCLK clock
// must be divided by 2
Modify CHIP_CLK_TOP_CTRL->INPUT_FREQ_DIV2 0x0001 // bit 3
Sys_MCLK_Input_Freq = Sys_MCLK_Input_Freq/2;
// PLL output frequency is different based on the sample clock
// rate used.
if (Sys_Fs_Rate == 44.1 kHz)
  PLL_Output_Freq = 180.6336 MHz
else
  PLL_Output_Freq = 196.608 MHz
// Set the PLL dividers
Int_Divisor = floor(PLL_Output_Freq/Sys_MCLK_Input_Freq)
Frac_Divisor = ((PLL_Output_Freq/Sys_MCLK_Input_Freq) - Int_Divisor)*2048
Modify CHIP_PLL_CTRL->INT_DIVISOR Int_Divisor // bits 15:11
Modify CHIP_PLL_CTRL->FRAC_DIVISOR Frac_Divisor // bits 10:0
```

Input/Output Routing

To avoid any pops/clicks, the outputs should be muted during these chip configuration steps. Refer to Volume Control for volume and mute control.

A few example routes are shown below:

```
// Example 1: I2S_IN -> DAP -> DAC -> LINEOUT, HP_OUT
// Route I2S_IN to DAP
Modify CHIP_SSS_CTRL->DAP_SELECT 0x0001 // bits 7:6
// Route DAP to DAC
Modify CHIP_SSS_CTRL->DAC_SELECT 0x0003 // bits 5:4
// Select DAC as the input to HP_OUT
Modify CHIP_ANA_CTRL->SELECT_HP 0x0000 // bit 6
// Example 2: MIC_IN -> ADC -> I2S_OUT
// Set ADC input to MIC_IN
Modify CHIP_ANA_CTRL->SELECT_ADC 0x0000 // bit 2
// Route ADC to I2S_OUT
Modify CHIP_SSS_CTRL->I2S_SELECT 0x0000 // bits 1:0
// Example 3: LINEIN -> HP_OUT
// Select LINEIN as the input to HP_OUT
Modify CHIP_ANA_CTRL->SELECT_HP 0x0001 // bit 6
```

DIGITAL AUDIO PROCESSOR CONFIGURATION

To avoid any pops/clicks, the outputs should be muted during these chip configuration steps. Refer to Volume Control for volume and mute control.

```
// Enable DAP block
// NOTE: DAP will be in a pass-through mode if none of DAP
// sub-blocks are enabled.
Modify DAP_CONTROL->DAP_EN 0x0001 // bit 0
```

Dual Input Mixer

These programming steps are needed only if dual input mixer feature is used.

```
// Enable Dual Input Mixer
Modify DAP_CONTROL->MIX_EN 0x0001 // bit 4
// NOTE: This example assumes mix level of main and mix
// channels as 100% and 50% respectively
// Configure main channel volume to 100% (No change from input
// level)
Write DAP_MAIN_CHAN 0x4000
// Configure mix channel volume to 50% (attenuate the mix
// input level by half)
Write DAP_MIX_CHAN 0x4000
```

Freescale Surround

The Freescale Surround on/off function will be typically controlled by the end-user. End-user driven programming steps are shown in End-user Driven Chip Configuration.

The default WIDTH_CONTROL of 4 should be appropriate for most applications. This optional programming step shows how to configure a different width value.