DIGITAL AUDIO PROCESSOR WITH ANALOG INTERFACE

Check for Samples: TAS3208

FEATURES

- **Digital Audio Processor**
- **Fully Programmable With Graphical, Drag-and-Drop PurePath Studio™ Software Development Environment**
- **135-MHz Operation**
- **48-Bit Data Path With 76-Bit Accumulator**
- **Hardware Single-Cycle Multiplier (28 × 48)**
- **Five Simultaneous Operations Per Clock Cycle**
- **Usable 768 Words Data RAM (48 Bit), Usable 1K Coefficient RAM (28 Bit)**
- **Usable 2.5K Program RAM**
- **360 ms at 48 kHz, 17K Words 24-Bit Delay Memory**
- **Slave Mode F_s is 44.1 kHz and 48 kHz**
- **Master Mode F_s is 48 kHz**
- **Analog Audio Input/Output**
  - **10:1 Stereo Analog Input MUX**
  - **Stereo Analog Pass-Through Channel**
  - **Stereo, Single-Ended ADC (93 dB DNR Typical)**
  - **Six Single-Ended DACs (97 dB DNR Typical)**
  - **Stereo Headphone Amplifier, 24-mW Power Output into 16 Ω, 100 pF**
- **Digital Audio Input/Output**
  - **Three Synchronous Serial Audio Inputs (Six Channels)**
  - **Two Synchronous Serial Audio Outputs (Four Channels)**
  - **Input and Output Data Formats: 16-, 20-, or 24-Bit Data Left, Right, and I^2S**
  - **SPDIF Transmitter**
- **System Control Processor**
  - **Embedded 8051 WARP Microprocessor**
  - **Programmable Using Standard 8051 C Compilers**
  - **Four Programmable GPIO Pins**
- **General Features**
  - **Two I^2C Ports for Slave or Master Download**
  - **Single 3.3-V Power Supply**
  - **Integrated Regulators**

APPLICATIONS

- **Flat-Screen TVs**
- **MP3 Players/Music Phone Docks**
- **Speaker Bars**
- **Mini/Micro Component Systems**
- **Automotive Head Units**
- **Musical Instruments**

DESCRIPTION

The TAS3208 is a highly-integrated audio system-on-chip (SOC) consisting of a fully-programmable 48-bit digital audio processor, 10:1 stereo analog input MUX, stereo ADC, six DACs, and other analog functionality. The TAS3208 is programmable with the graphical PurePath Studio™ suite of DSP code development software. PurePath Studio is a highly intuitive, drag-and-drop development environment that minimizes software development effort while allowing the end user to utilize the power and flexibility of the TAS3208's digital audio processing core.

TAS3208 processing capability includes speaker equalization and cross over, volume/bass/treble control, signal mixing/MUXing/splitting, delay compensation, dynamic range compression, and many other basic audio functions. Audio functions such as matrix decoding, stereo widening, surround sound virtualization and psychoacoustic bass boost are also available with either third party or TI royalty-free algorithms.

The TAS3208 contains a custom-designed, fully-programmable 135-MHz, 48-bit digital audio processor. A 76-bit accumulator ensures that the high precision necessary for quality digital audio is maintained during arithmetic operations. A stereo, 93-dB DNR ADC and six 97-dB DNR DACs ensure that high-quality audio is maintained through the whole signal chain.

The TAS3208 is composed of seven functional blocks:

- Clock and serial data interface
- Analog input and output
- M8051 WARP controller, serial control interface, and device control
- Audio DSP – digital audio processing
- Power supply
- Internal references

Figure 1 shows the functional structure of the TAS3208.

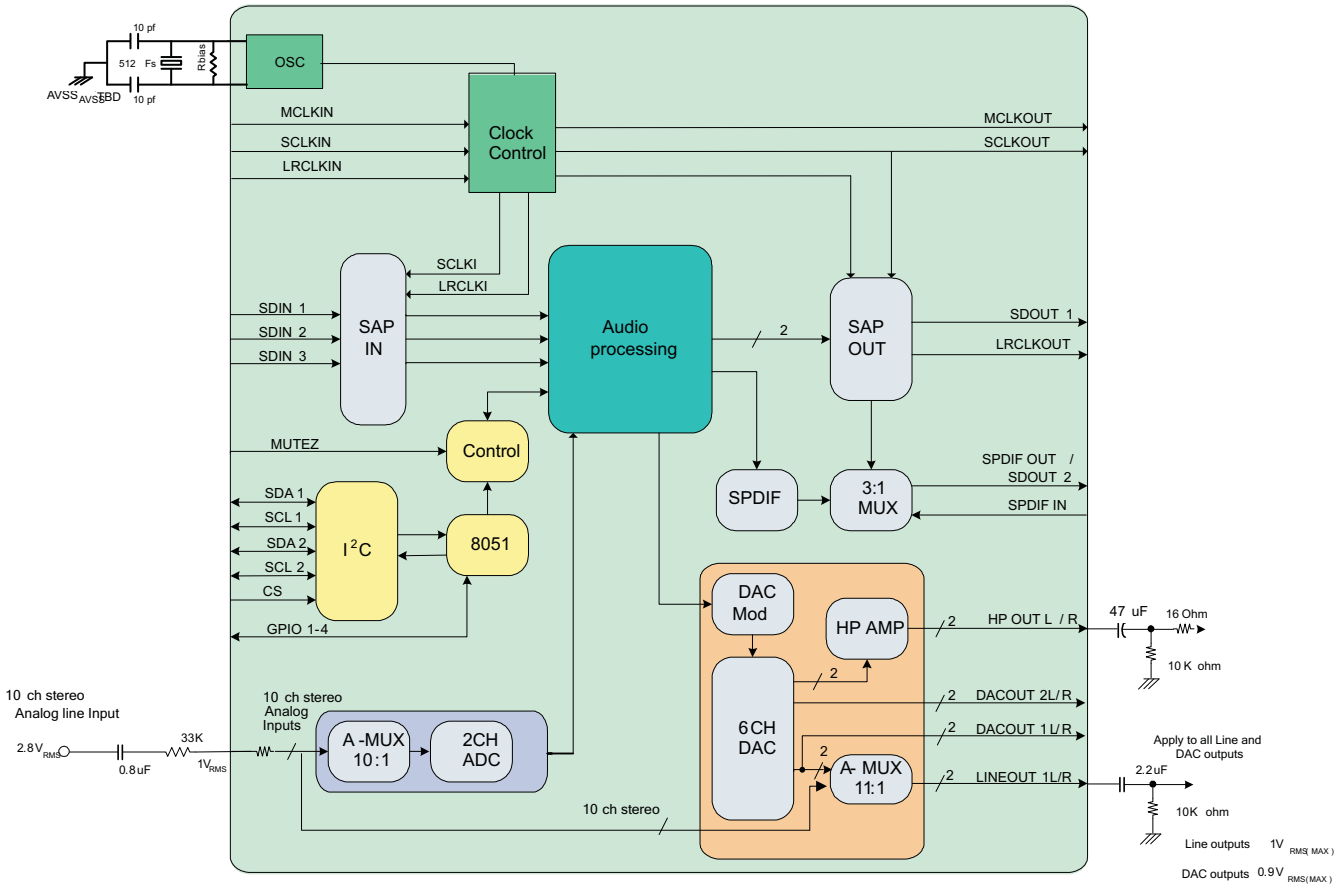


Figure 1. Block Diagram

The TAS3208 may be used with an external asynchronous sample rate converter (ASRC) to accommodate asynchronous serial inputs at different sampling rates (see Figure 2).

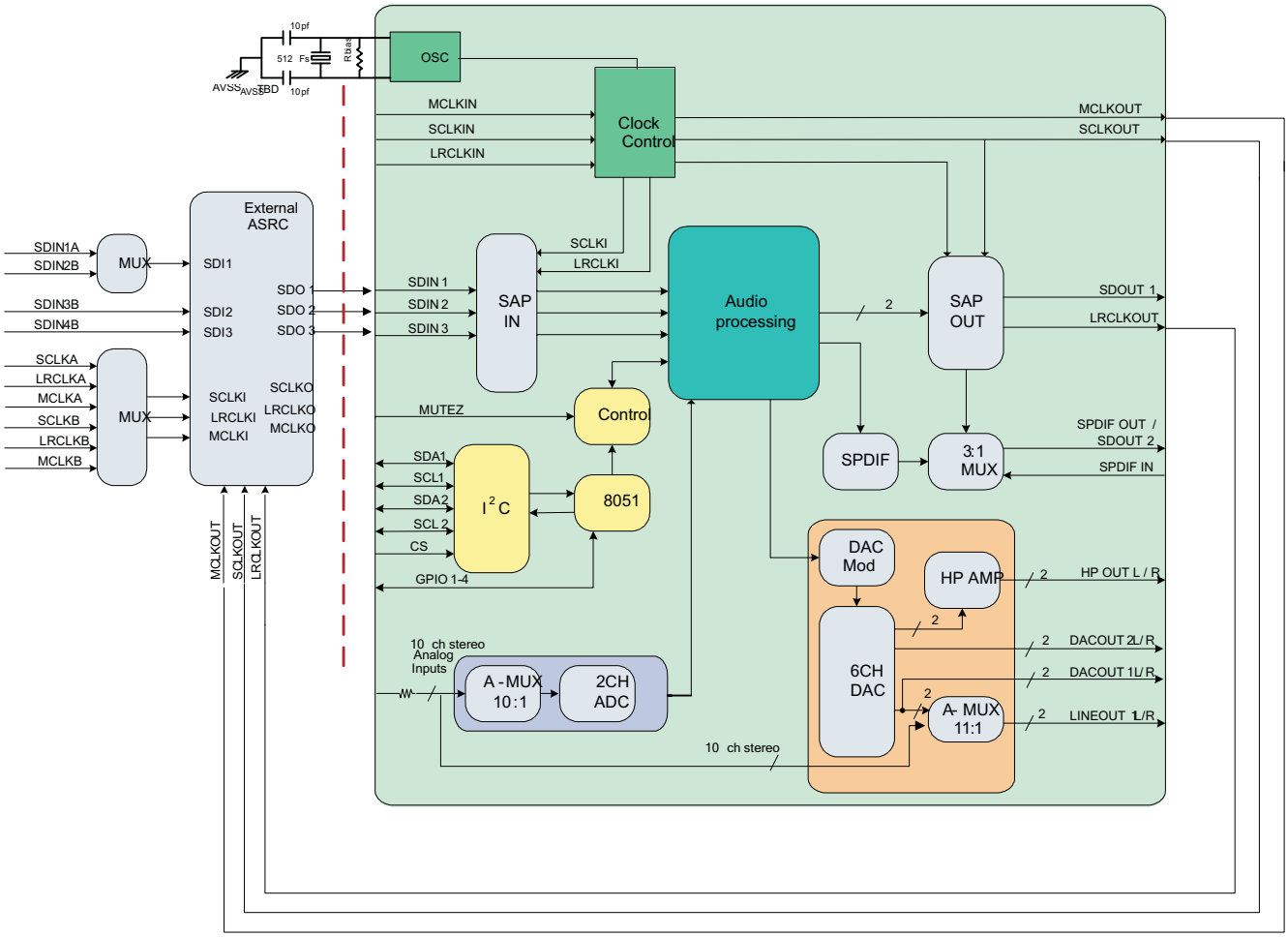


Figure 2. Interface to External ASRC

PZP PACKAGE
(TOP VIEW)

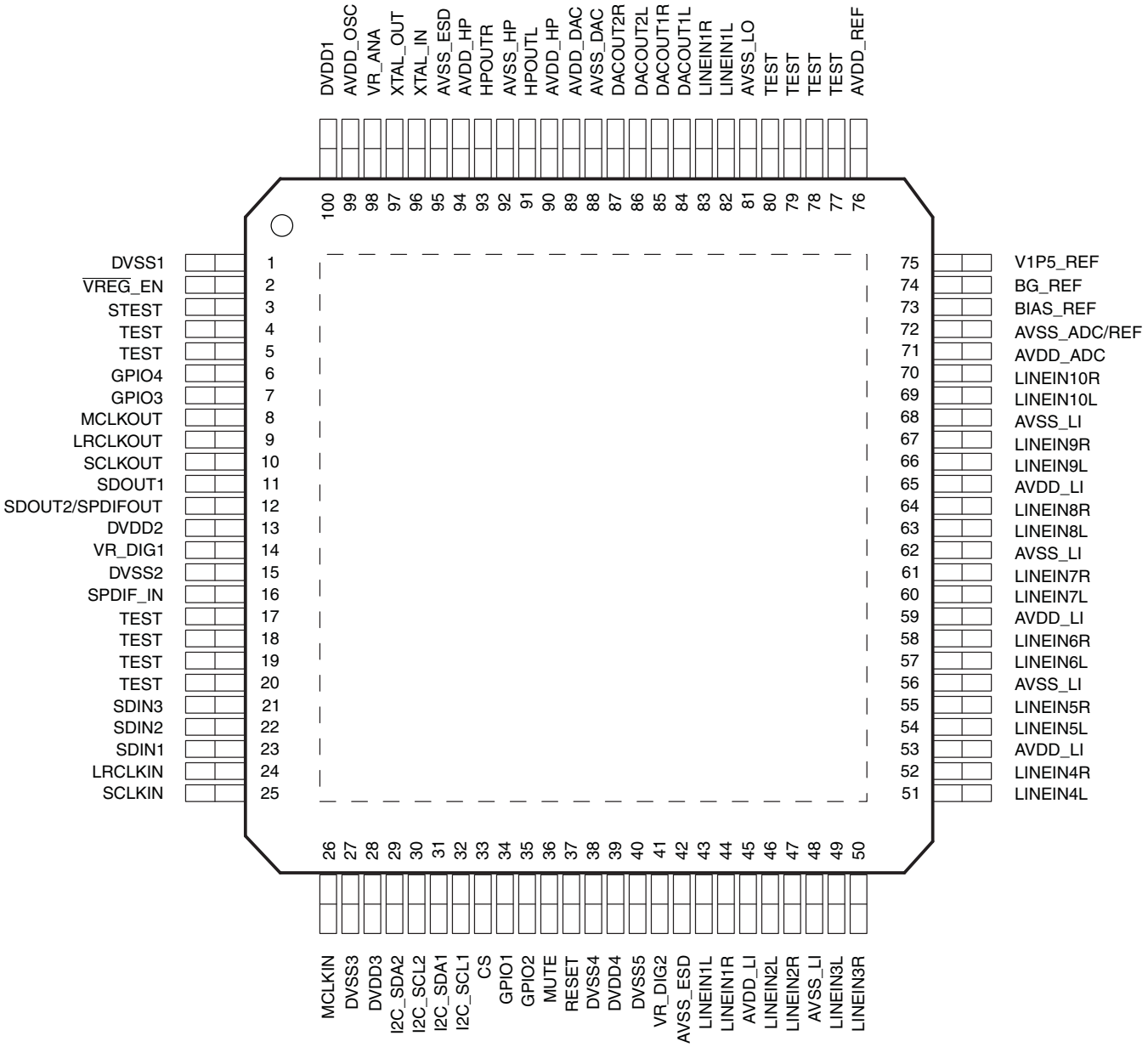


Table 1. ORDERING INFORMATION

T_A	PACKAGE(1) (2)		ORDERABLE PART NUMBER	TOP-SIDE MARKING
–40°C to 85°C	TQFP – PZP	Tape and reel	TAS3208IPZP	TAS3208IPZP
			TAS3208IPZPR	
–20°C to 70°C			TAS3208PZP	TAS3208PZP
			TAS3208PZPR	

(1) Package drawings, thermal data, and symbolization are available at www.ti.com/packaging.
(2) For the most current package and ordering information, see the Package Option Addendum at the end of this document, or see the TI website at www.ti.com.

Table 2. TERMINAL FUNCTIONS

TERMINAL		I/O	TERMINATION(1)	DESCRIPTION
NO.	NAME			
1	DVSS1	P		Digital ground
2	$\overline{\text{VREG}}$_EN	DI		Voltage regulator enable
3	STEST	DI	Pulldown	Test pin to reconfigure pins
4, 5, 17, 18, 19, 20	TEST	–	Pulldown	
6	GPIO4	DIO	Pulldown	General-purpose input/output 4
7	GPIO3	DIO	Pulldown	General-purpose input/output 3
8	MCLKOUT	DO		Master clock output
9	LRCLKOUT	DO		Left/right (frame) clock output
10	SCLKOUT	DO		Serial audio data clock output
11	SDOUT1	DO		Serial digital audio data output 1
12	SDOUT2/ SPDIF_OUT	DO		Serial digital audio data out 2/SPDIF output
13	DVDD2	P		3.3-V digital power
14	VR_DIG1	P		Pinout of internal regulator. A 4.7-µF low-ESR capacitor should be connected between this pin and digital ground. This terminal must not be used to power external devices.
15	DVSS2	P		Digital ground
16	SPDIF_IN	DI		SPDIF input
21	SDIN3	DI		Serial digital audio data input 3
22	SDIN2	DI		Serial digital audio data input 2
23	SDIN1	DI		Serial digital audio data input 1
24	LRCLKIN	DI		Left/right (frame) clock input
25	SCLKIN	DI		Serial audio data clock input
26	MCLKIN	DI		Master clock input
27	DVSS3	P		Digital ground
28	DVDD3	P		3.3-V digital power master
29	I2C_SDA2	DIO		I^2C serial data master
30	I2C_SCL2	DIO		I^2C serial clock slave
31	I2C_SDA1	DIO		I^2C serial data slave
32	I2C_SCL1	DIO		I^2C serial clock
33	CS	DI		Chip select
34	GPIO1	DIO		General-purpose input/output 1
35	GPIO2	DIO		General-purpose input/output 2
36	$\overline{\text{MUTE}}$	DI	Pullup	Mute device
37	$\overline{\text{RESET}}$	DI	Pullup	Reset
38	DVSS4	P		Digital ground
39	DVDD4	P		3.3-V digital power
40	DVSS5	P		3.3-V digital power
41	VR_DIG2	P		Pinout of internal regulator. A 4.7-µF low-ESR capacitor should be connected between this pin and digital ground. This terminal must not be used to power external devices.
42	AVSS_ESD	P		Analog ESD ground
43	LINEIN1L	AI		Left-channel analog input 1

(1) All pullups are 20-µA weak pullups, and all pulldowns are 20-µA weak pulldowns (166 kΩ) . The pullups and pulldowns are included to ensure proper input logic levels if the terminals are left unconnected (pullups at logic 1 input; pulldowns at logic 0 input). Devices that drive inputs with pullups must be able to sink 20 µA while maintaining a logic 0 drive level. Devices that drive inputs with pulldowns must be able to source 20 µA while maintaining a logic 1 drive level.

Table 2. TERMINAL FUNCTIONS (continued)

TERMINAL		I/O	TERMINATION(1)	DESCRIPTION
NO.	NAME			
44	LINEIN1R	AI		Right-channel analog input 1
45, 53, 59, 65	AVDD_LI	P		3.3-V analog power
46	LINEIN2L	AI		Left-channel analog input 2
47	LINEIN2R	AI		Right-channel analog input 2
48, 56, 62, 68	AVSS_LI	P		Analog ground
49	LINEIN3L	AI		Left-channel analog input 3
50	LINEIN3R	AI		Right-channel analog input 3
51	LINEIN4L	AI		Left-channel analog input 4
52	LINEIN4R	AI		Right-channel analog input 4
54	LINEIN5L	AI		Left-channel analog input 5
55	LINEIN5R	AI		Right-channel analog input 5
57	LINEIN6L	AI		Left-channel analog input 6
58	LINEIN6R	AI		Right-channel analog input 6
60	LINEIN7L	AI		Left-channel analog input 7
61	LINEIN7R	AI		Right-channel analog input 7
63	LINEIN8L	AI		Left-channel analog input 8
64	LINEIN8R	AI		Right-channel analog input 8
66	LINEIN9L	AI		Left-channel analog input 9
67	LINEIN9R	AI		Right-channel analog input 9
69	LINEIN10L	AI		Left-channel analog input 10
70	LINEIN10R	AI		Right-channel analog input 10
71	AVDD_ADC	P		3.3-V analog power
72	AVSS_ADC/REF	P		Analog ground
73	BIAS_REF	AO		Pin should be tied to analog ground with 22 kΩ ± 1%.
74	BG_REF	AO		Band-gap output. Must be tied to ground with 1-µF low-ESR capacitor.
75	V1P5_REF	AO		Common-mode output. Must be tied to ground with 1-µF low-ESR capacitor.
76	AVDD_REF	P		3.3-V analog power
77, 78, 79, 80	TEST	–		
81	AVSS_LO	P		Analog ground
82	LINEOUT1L	AO		Left-channel analog output 1
83	LINEOUT1R	AO		Right-channel analog output 1
84	DACOUT1L	AO		Left-channel digital-to-analog converter output 1
85	DACOUT1R	AO		Right-channel digital-to-analog converter output 1
86	DACOUT2L	AO		Left-channel digital-to-analog converter output 2
87	DACOUT2R	AO		Right-channel digital-to-analog converter output 2
88	AVSS_DAC	P		Analog ground
89	AVDD_DAC	P		3.3-V analog power
90	AVDD_HP	P		3.3-V analog power
91	HPOUTL	AO		Left-channel headphone output
92	AVSS_HP	P		Analog ground
93	HPOUTR	AO		Right-channel headphone output
94	AVDD_HP	P		3.3-V analog power
95	AVSS_ESD	P		Analog ground
96	XTAL_IN	DI		External crystal input

Table 2. TERMINAL FUNCTIONS (continued)

TERMINAL		I/O	TERMINATION(1)	DESCRIPTION
NO.	NAME			
97	XTAL_OUT	DO		External crystal output
98	VR_ANA	P		Pinout of internal regulator. A 4.7-µF low-ESR capacitor should be connected between this pin and digital ground. This terminal must not be used to power external devices.
99	AVDD_OSC	P		3.3-V analog power
100	DVDD1	P		3.3-V digital power

Clocks

The TAS3208 can be configured as either the clock master or clock slave depending on the settings in the clock configuration register. By default, the TAS3208 is configured as the clock master. Figure 3 shows the block diagram of the TAS3208 clocks.

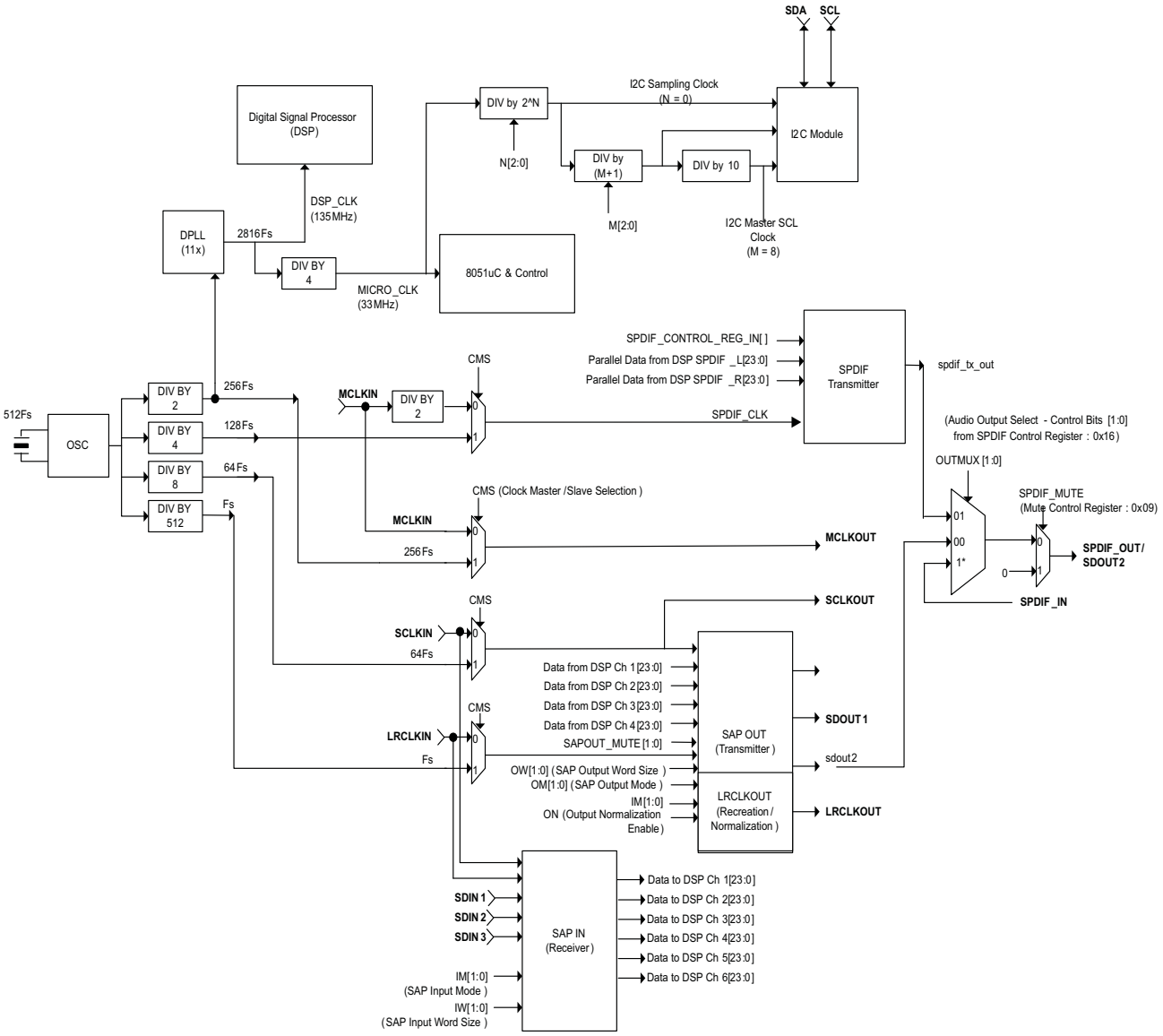


Figure 3. Clocking System

Digital Audio Interface

The TAS3208 has three digital inputs that accept discrete I^2S, discrete left-justified, and discrete right-justified PCM data.

The TAS3208 has two digital outputs that provide discrete I^2S, discrete left-justified, and discrete right-justified PCM data.The second digital output can also be configured to provide SPDIF encoded PCM data.

The TAS3208 has a SPDIF input that is capable of routing an SPDIF-encoded signal through the device. This input is not processed by the digital audio processor (DAP). The clocking system for the device is shown in Figure 4.

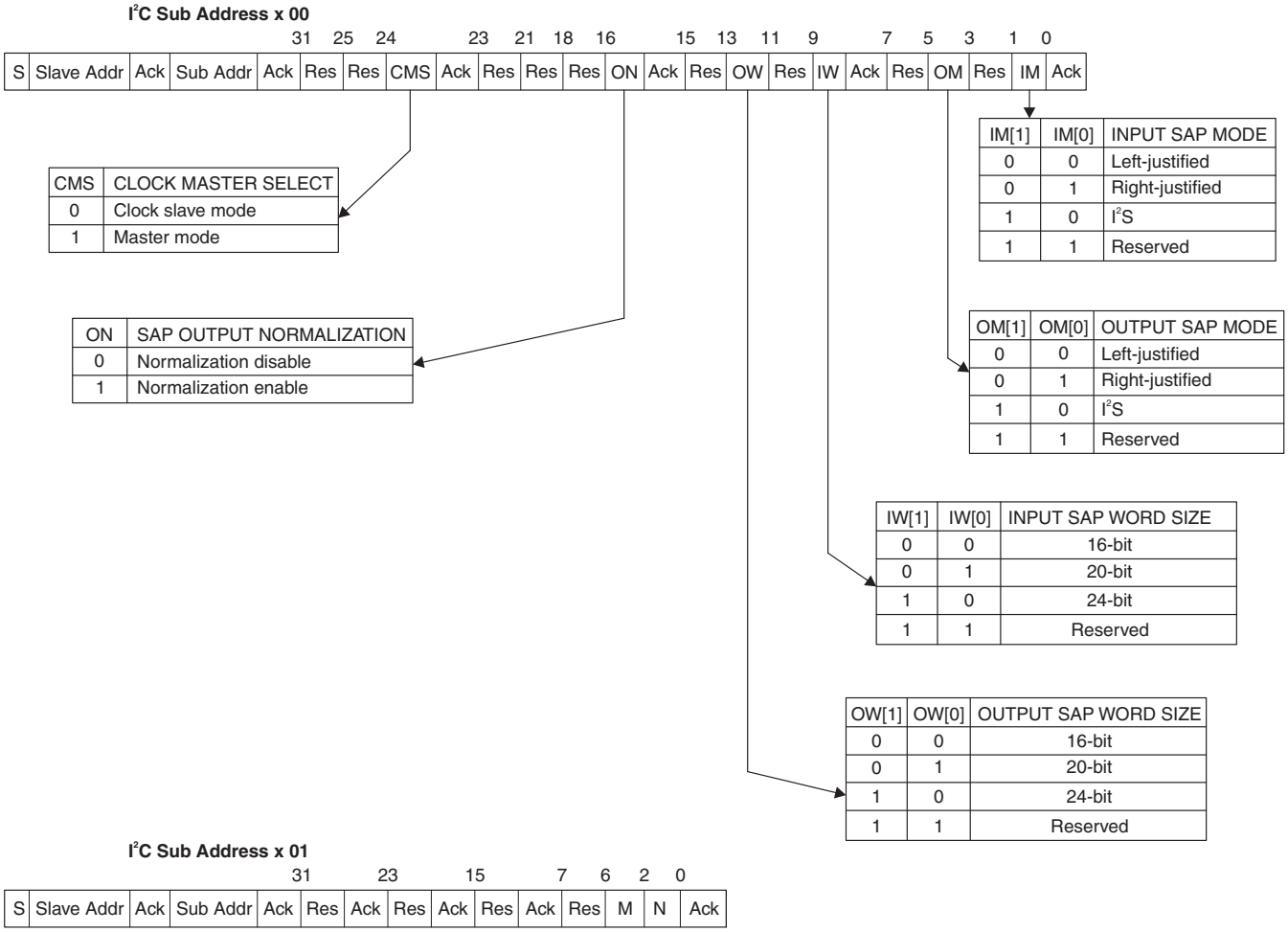


Figure 4. Clocking System I²C Mapping

Clock Master Operation

When configured as the device clock master, an external crystal is used as a reference to an internal oscillator. In this mode of operation, all internal clocks are generated by the oscillator.

- LRCLKOUT is fixed at 48 kHz (Fs).
- SCLKOUT is fixed at 64 × Fs.
- MCLKOUT is fixed 256 × Fs.

Clock Slave Operation

When configured as the device clock slave, the DAP, MCU, and I²C interface are derived from the external crystal. However, the digital audio clocks are supplied externally.

Internal analog clocks for the analog-to-digital converter (ADC) and digital-to-analog converter (DAC) are derived from the MCLKIN input. As a result, analog performance depends on the quality of MCLKIN.

Degradation in analog performance is to be expected, depending on the quality of MCLKIN.

The TAS3208 device does not include any internal clock error or click/pop detection/management. The muting of the outputs at updating of sample-rate-dependent coefficients must be initiated by the host system controller.

MCLKOUT, SCLKOUT, and LRCLKOUT are passed through from the clock inputs MCLKIN, SCLKIN, and LCLKIN.

- MCLKIN 256 × Fs is supplied externally.
- SCLKIN 64 × Fs is supplied externally.
- LRCLKIN Fs is supplied externally.

NOTE

In slave mode, all incoming serial audio data must be synchronous to an incoming LRCLKIN of 32, 44.1, or 48 kHz. The TAS3208 does not support the use of an external (i.e., 24-MHz) clock input into XTALI.

Digital Audio Data Formats

Serial data is input on pins SDIN3–SDIN1 on the TAS3208, allowing up to six channels of digital audio input. The TAS3208 supports 16-, 20-, or 24-bit data in left, right, or I^2S serial data format. By default, all TAS3208 serial digital inputs are configured in the 24-bit I^2S format. The serial data input format is configurable via the SAP/Clock Settings register.

Serial data is output on pins SDOUT1 and SDOUT2, allowing up to four channels of digital audio output. By default, the SDOUT data format is 24-bit I^2S format at the same data rate as the input. The SDOUT1 and SDOUT2 outputs use SCLKOUT and LRCLKOUT signals to provide synchronization. SDOUT2 is multiplexed with an SPDIF output.

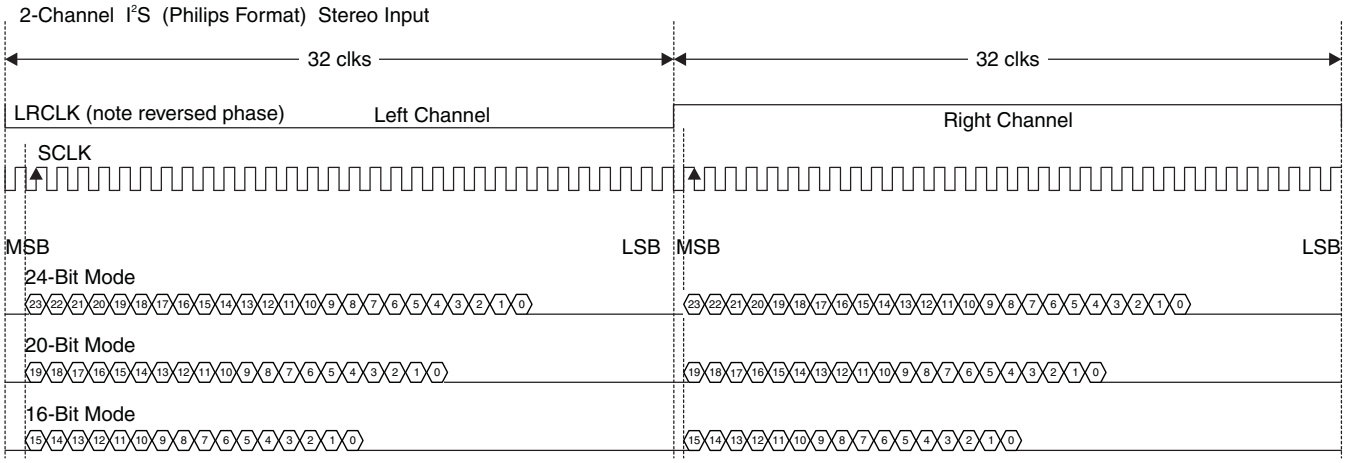
NOTE

To avoid audio artifacts, I^2C commands to reconfigure the serial audio port (SAP) should not be issued as stand-alone commands, rather they should be accompanied by mute and unmute commands.

The TAS3208 uses the SCLK as a reference for both input and output samples. The negative edge of SCLK is used to output a new data bit, whereas the positive edge of SCLK is used to sample incoming serial data.

Discrete I^2S Timing

I^2S timing uses an LRCLK to define when the data being transmitted is for the left channel and when it is for the right channel. The LRCLK is LOW for the left channel and HIGH for the right channel. A bit clock running at 64 × Fs is used to clock in the data. There is a delay of one bit clock from the time the LRCLK signal changes state to the first bit of data on the data lines. The data is written most significant bit (MSB) first and is valid on the rising edge of bit clock. The TAS3208 will mask unused trailing data bit positions.

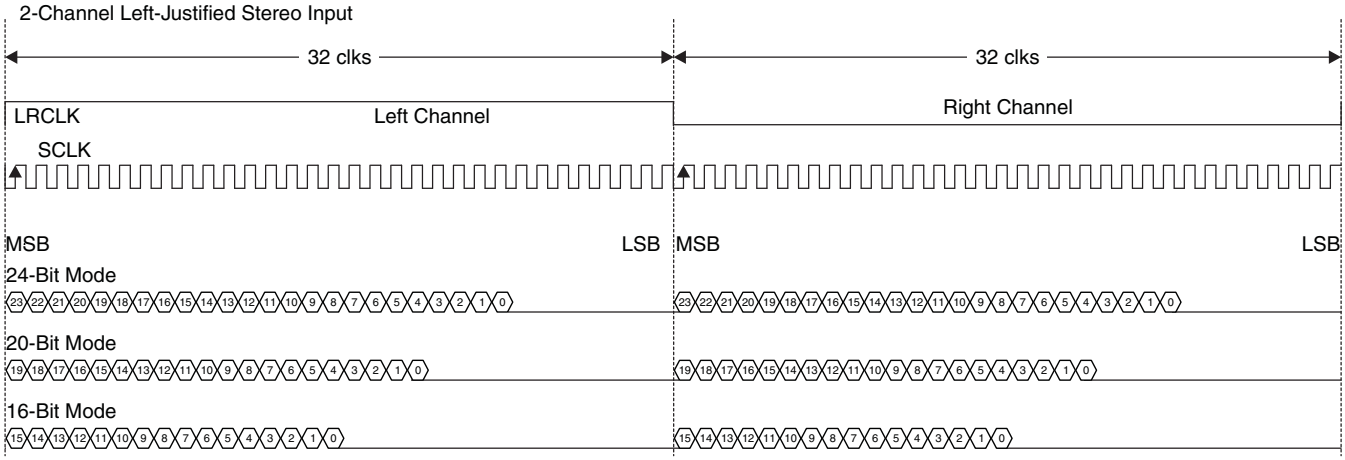


A. All data are presented in 2s-complement form with MSB first.

Figure 5. SAP I^2S 64 × Fs Format

Discrete Left-Justified (LJ) Timing

Left-justified timing uses an LRCLK to define when the data being transmitted is for the left channel or right channel. The LRCLK is HIGH for the left channel and LOW for the right channel. A bit clock running at 64 × Fs is used to clock in the data. The first bit of data appears on the data lines at the same time the LRCLK toggles. The data is written MSB first and is valid on the rising edge of bit clock. The TAS3208 will mask unused trailing data bit positions.

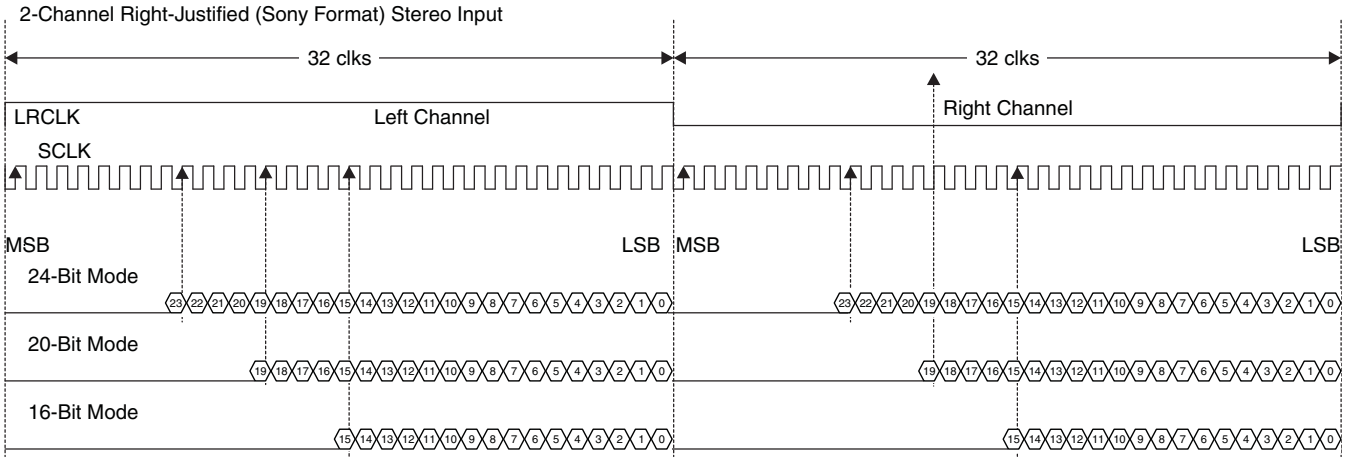


A. All data are presented in 2's complement form with MSB first.

Figure 6. SAP Left-Justified 64 × Fs Format

Discrete Right-Justified (RJ) Timing

Right-justified timing uses an LRCLK to define when the data being transmitted is for the left channel or right channel. The LRCLK is HIGH for the left channel and LOW for the right channel. A bit clock running at 64 × Fs is used to clock in the data. The first bit of data appears on the data 8-bit clock periods (for 24-bit data) after L/RCLK toggles. In RJ mode, the LSB of data is always clocked by the last bit clock before L/RCLK transitions. The data is written MSB first and is valid on the rising edge of bit clock. The TAS3208 will mask unused leading data bit positions.



A. All data are presented in 2s-complement form with MSB first.

Figure 7. SAP Right-Justified 64 × Fs Format

SAP Input and Output Normalization

The TAS3208 supports SAP input and SAP output normalization. This supports simultaneous output to left-justified and I^2S devices.

NOTE

The normalization function is only available in slave mode.

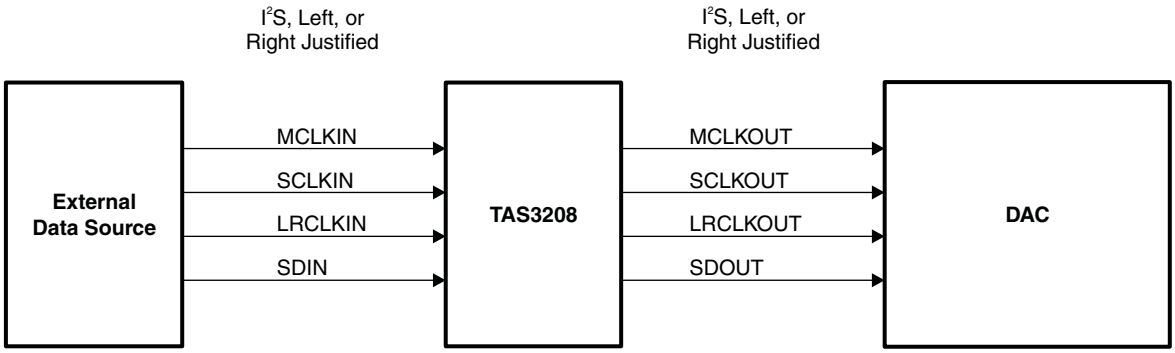


Figure 8. SAP Output Normal Configuration (No Normalization)

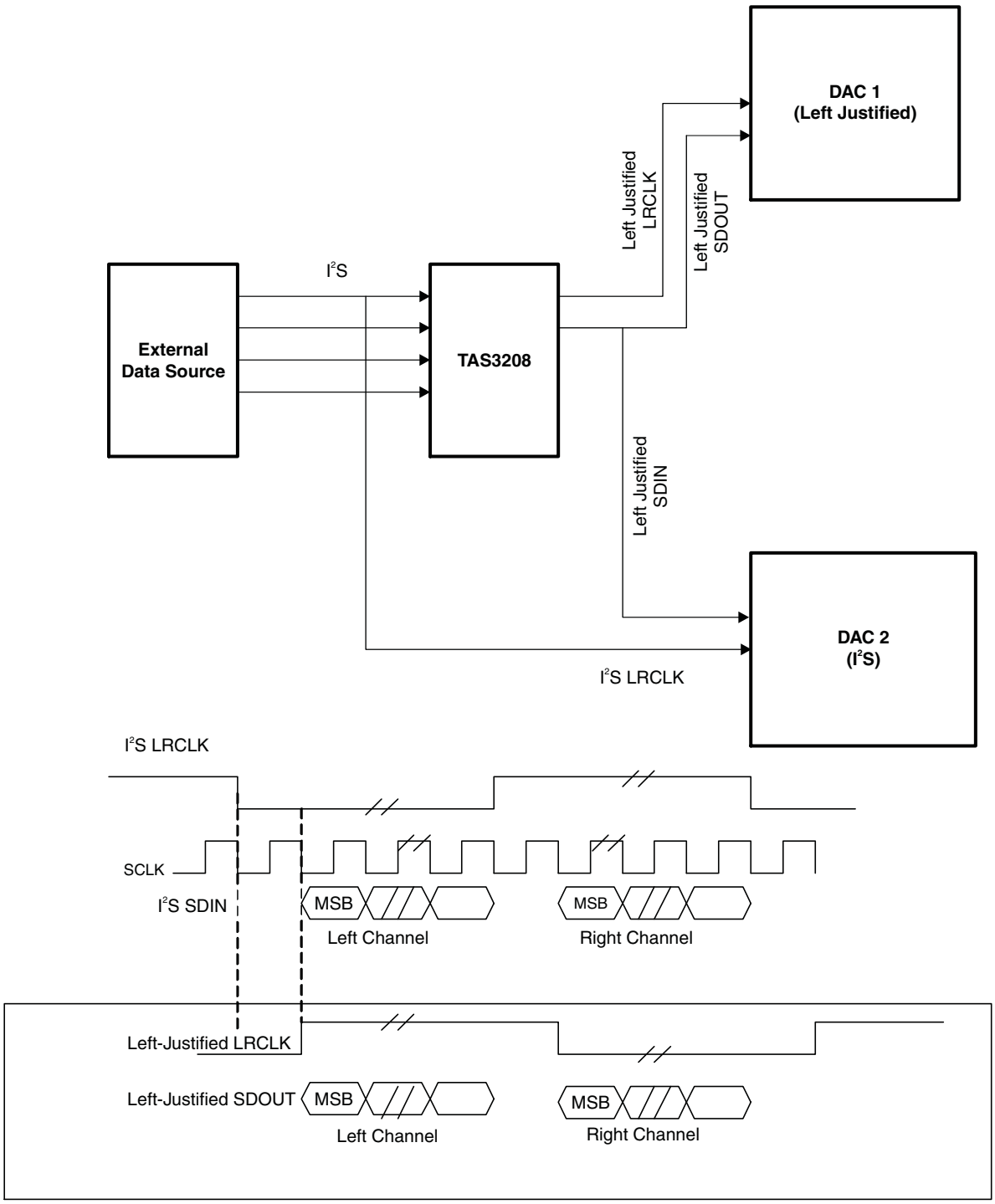


Figure 9. SAP Output Configuration (I^2S to Left Normalization ON)

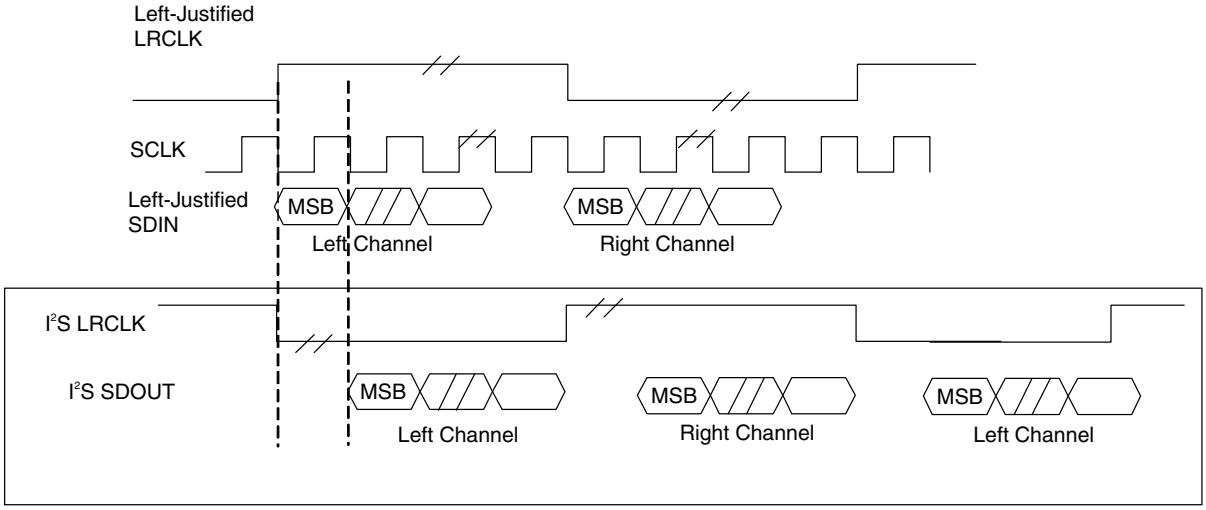


Figure 10. SAP Output Configuration (I^2S to Left Normalization OFF)

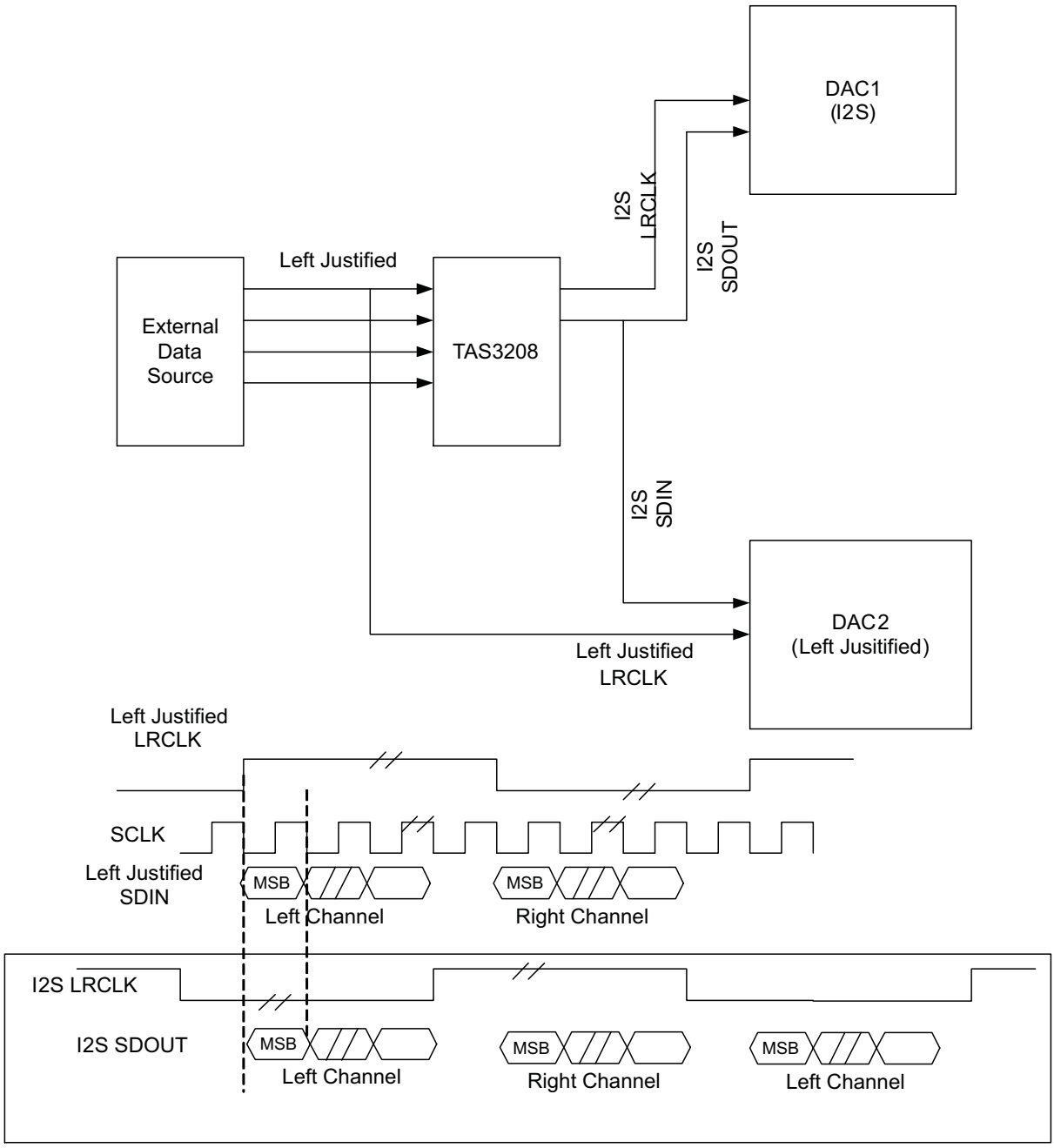


Figure 11. SAP Output Configuration (Left to I^2S Normalization ON)

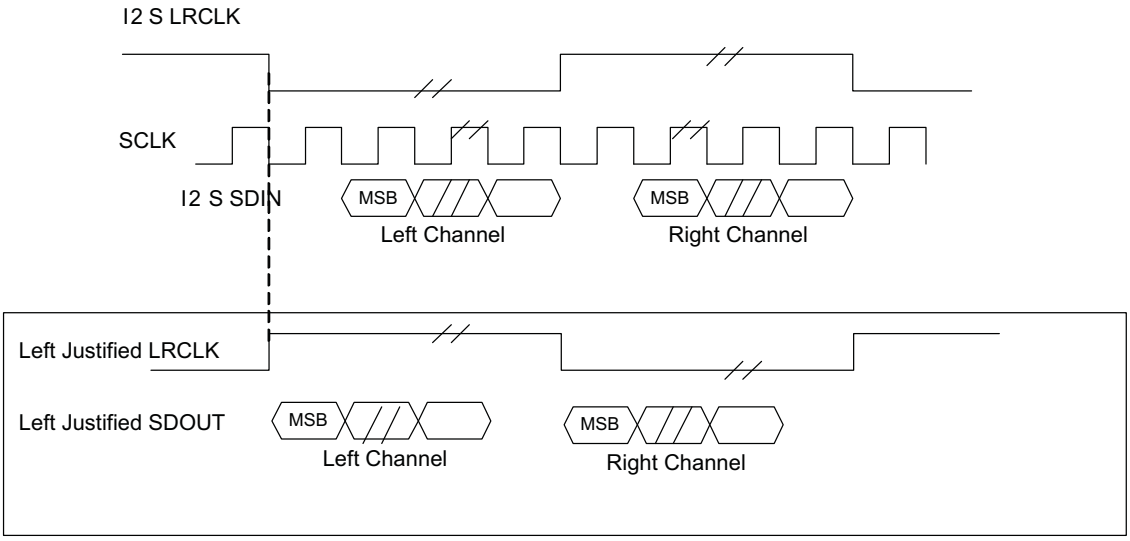


Figure 12. SAP Output Configuration (Left to I²S Normalization OFF)

SPDIF Encoder

The SPDIF encoder is a digital audio transmitter designed for use in consumer audio applications. Transmit data rates up to 48 kHz are supported. The SPDIF encoder complies with the IEC 60958 interface standard.

The SPDIF encoder creates a multiplexed bit stream containing audio, status, and user data. The multiplexed data format is shown in Figure 14. The data is then biphase mark encoded and output.

The hardware architecture of the SPDIF encoder is shown in Figure 13.

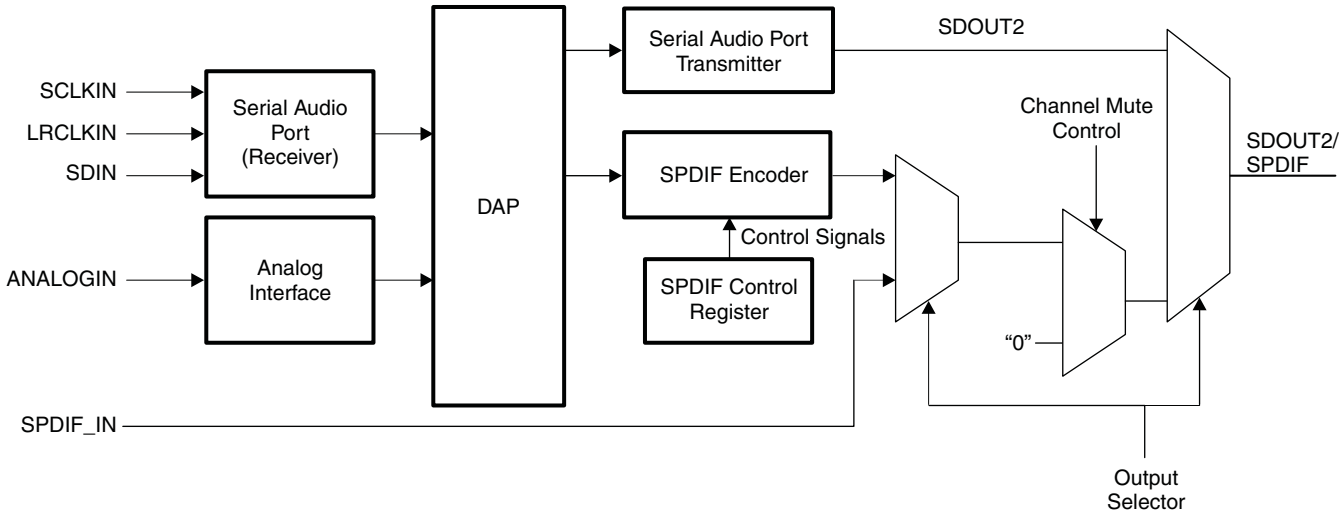


Figure 13. SPDIF Encoder Hardware Architecture

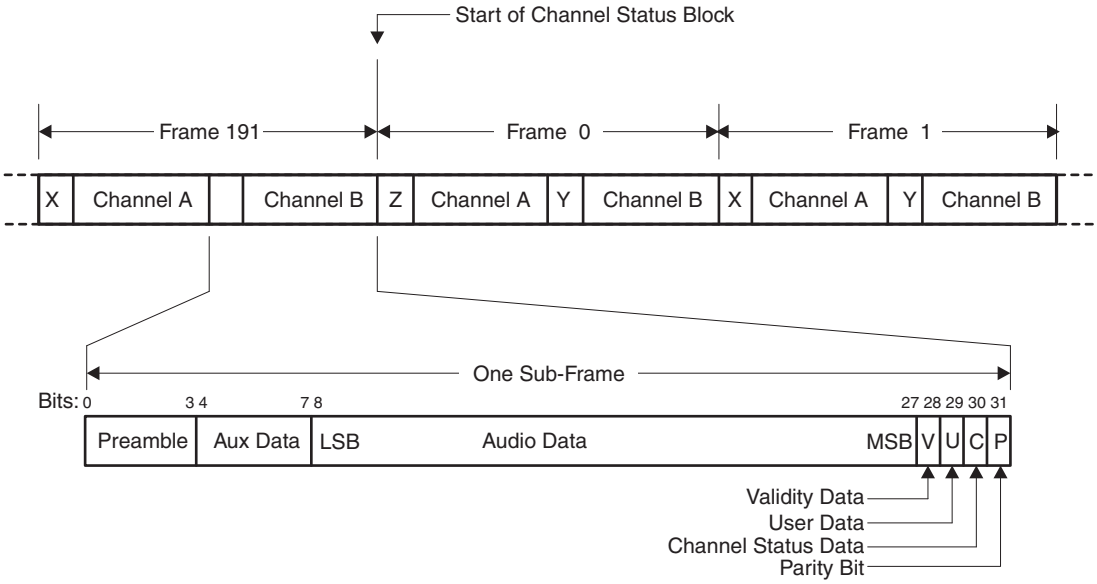


Figure 14. SPDIF Frame Format

SPDIF Encoder Operation

The SPDIF encoder performs the multiplexing of audio, channel status, user, and validity flag. It also performs biphase mark encoding of the multiplexed data stream. Audio data for both left and right channels from the DAP are latched at the rising edge of the internal LRCLK, which marks the beginning of next sample cycle. The SPDIF encoder then multiplexes these samples with internally-generated preambles, channel status, user data, validity flag, and parity. The channel status and validity flag are generated based on the settings in the SPDIF control registers, while the user data is fixed to all zero. The biphase mark-encoded signal is then output starting at the next rising edge of the internal LRCLK. The generated SPDIF stream is fixed to consumer-mode linear audio PCM format.

While the $\overline{\text{RESET}}$ input is low, the transmitter output (SPDIF_OUT) is forced to logic low level. Upon setting $\overline{\text{RESET}}$ high, the SPDIF encoder remains inactive until the module reset is removed by writing 0 to the RST bit of the control register. Then this module will wait for synchronization with the internal frame clock and start encoding audio data. It is recommended to set all other SPDIF control register bits before releasing the module reset.

Transmitter Control Register

Table 3 shows the M8051 SFR register map for the SPDIF module control.

Table 3. M8051 SFR Register Map

ADDR	7	6	5	4	3	2	1	0
xx00	RST				CP			EMP
xx01	CATEGORY							L
xx10	SR		VL	VR	SRCNUM			
xx11			CLKAC		WORDLEN			

The relationship of the M8051 SFR register map with I^2C registers is described in Table 4.

Table 4. Relationship of M8051 SFR Register Map With I²C Registers

RST 0 1	Module reset Normal operation Reset SPDIF TX module (default)
CP 0 1	Copy permit Copy prohibit (default) Copy permit
EMP 0 1	Preemphasis No preemphasis (default) 50-/15-µs 2-channel preemphasis
CATEGORY	Category code 7-bit device category code (default “0101010”) (digital sound processor)
L 0 1	Generation status Generation 1 or higher (default) Original
SR 00 01 10 11	Sampling rate 44.1 kHz 48 kHz (default) Reserved 32 kHz
VL 0 1	Validity for left channel Left-channel data is valid (default) Left-channel data is invalid
VR 0 1	Validity for right channel Right-channel data is valid (default) Right-channel data is invalid
SRCNUM "0000" "0001" "0010" "0011" ⋮ "1000"	Source channel number Not specified 1 2 (default) 3 ⋮ 8
CLKAC "00" "01" "10" "11"	Clock accuracy Level II, 1000 ppm Level III, variable pitch shifted Level I, 50 ppm (default) Reserved
WORDLEN "0000: "0001" "0010" ⋮ "0100" ⋮ "1000" Others	Sample bit size 24 bits (default) 23 bits 22 bits ⋮ 20 bits ⋮ 16 bits Reserved

I²C Register Map for SPDIF

Figure 15 shows system-accessible I²C register mapping for controlling the SPDIF module. The mute control (MTE) uses the same control bits for controlling SDOUT2 mute at subaddress 0x09, and the module reset (RST) is mapped to subaddress 0x10 together with other power-down control bits. Other control bits are mapped to subaddress 0x16.

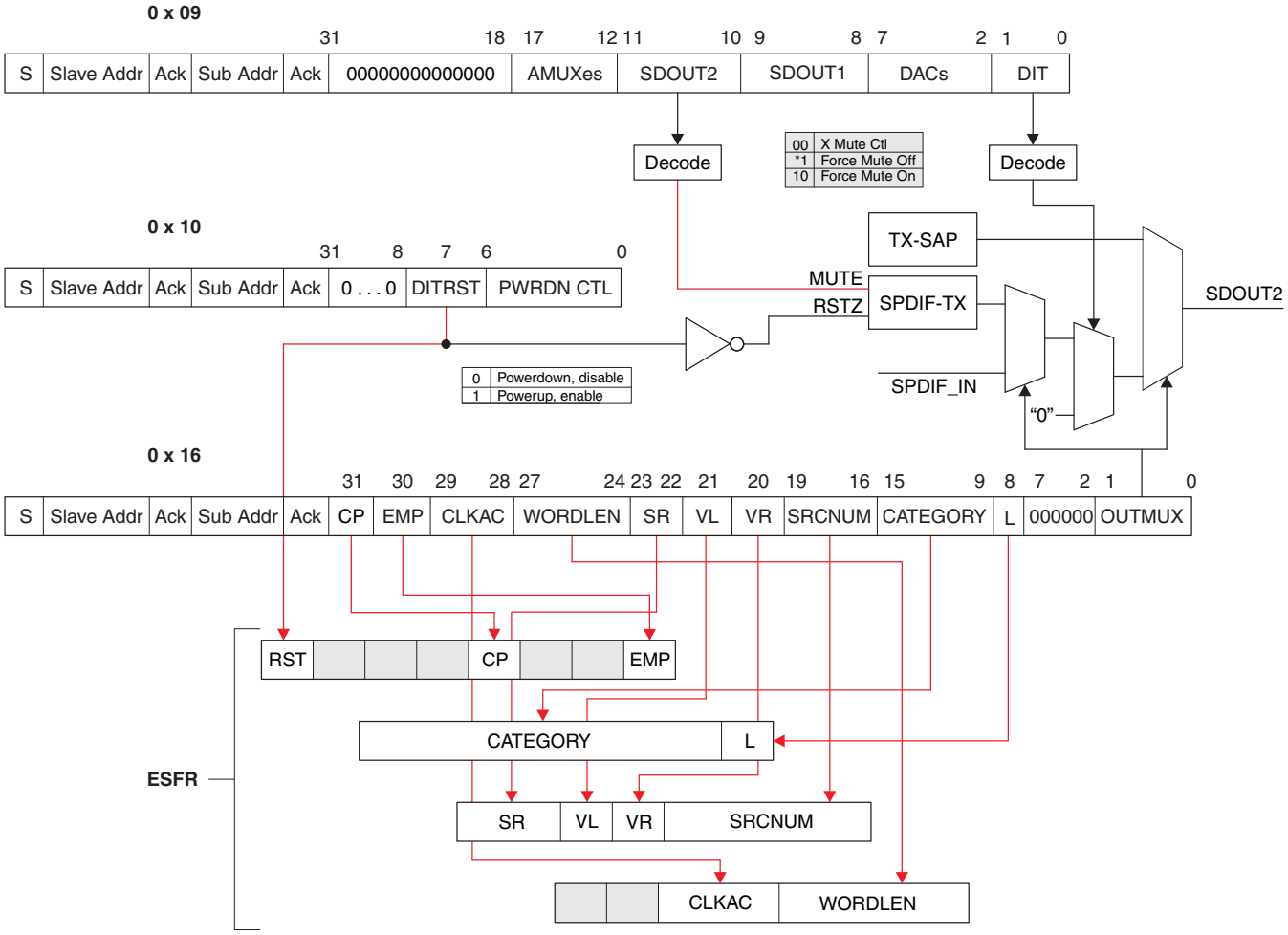


Figure 15. I²C Register to EFSR and Hardware Connection Map

Specification Coverage

The TAS3208 is covered by the following specifications:

- IEC 60956-1: Second Edition, 2004-03
- IEC 60956-3: Second Edition, 2003-01
- IEC 958-2: First Edition, 1994-07

Specification coverage details can be found in Table 5.

Table 5. TAS3208 Specification Coverage(1)

SPECIFICATION	SECTION	SUPPORTED	REMARKS
IEC 60958-1	Interface Format (4)	Yes	Auto frame formatting
	Channel Status (5)	Yes	First two bits fixed to 00 (consumer, linear PCM)
IEC 958-2	Mode 1 (software info delivery using b32–191 of channel stat) (4.2.2.1–4.2.2.3)	No	Bits 28–191 fixed to all zero
IEC 60958-3	Channel Status – General (5.1)	Yes	First channel status bit fixed to 0
	Channel Status – Application (5.2.1) – Byte0 (control)	Yes	b0–1: Fixed (00) b2: Register settable b3–5: Register settable b6–7: Fixed (00)
	Channel Status – Application (5.2.2) – Byte1 (category)	Yes, with restriction	Category code is register settable, with default value 0101010L (digital sound processor), but user data is fixed to all zero.
	Channel Status – Application (5.2.2) – Byte2 (source and channel number)	Yes	b16–19: Register settable b20–23: H/W auto set (1 for left, 2 for right channel)
	Channel Status – Application (5.2.2) – Byte3 (sampling freq and clock accuracy)	Yes, with restriction	b24–27: Register settable (32, 44.1, 48 kHz only) b28–29: Register settable
	Channel Status – Application (5.2.2) – Byte4 (word length, original sampling rate, Byte0, b1, 6, 7 = “0”)	Yes, partially	b32–35 : H/W auto set according to register setting, 24-bit original output sample truncated to the specified word length b36–39 : Fixed to all zero (not indicated)
	Category Code Groups (5.3.2)	Yes, with restriction	Specifying categories other than 0101010L (digital sound processor), especially those requiring nonzero user data is not recommended.
	User Data (6)	All zero	
	Timing Accuracy (7.2.1)	Yes	Clock accuracy indication is register settable. Expected to set level I (50 ppm) for master mode (XTAL source) or level II (1000 ppm) for slave mode.
	Line Driver Characteristics (7.3.2)	No	Standard output buffer. Needs external SPDIF driver (e.g., optical driver).

(1) Other sections of the specification not mentioned here are either considered irrelevant or covered elsewhere. IEC 60958-4 is specific for professional applications and, thus, irrelevant.

Analog Audio Interface

The TAS3208 is has ten analog stereo inputs that are multiplexed to one ADC. Additionally, the TAS3208 has one line output that can source any of the ten analog stereo inputs.

The TAS3208 has three stereo DACs. The outputs of of DAC3 are designed to be used as a 24-mW headphone amplifier or line driver. The other two DAC outputs are configured as stereo line drivers.

Both the ADC and DAC blocks can be placed in power down when not used.

Figure 16 shows a block diagram of the analog interface.

Stereo Analog-to-Digital Converter (ADC)

The TAS3208 has an analog 10:1 input multiplexer and an 11:1 output multiplexer. These can accept analog stereo inputs up to 1 Vrms. The outputs of the multiplexers are the stereo ADC and the line output.

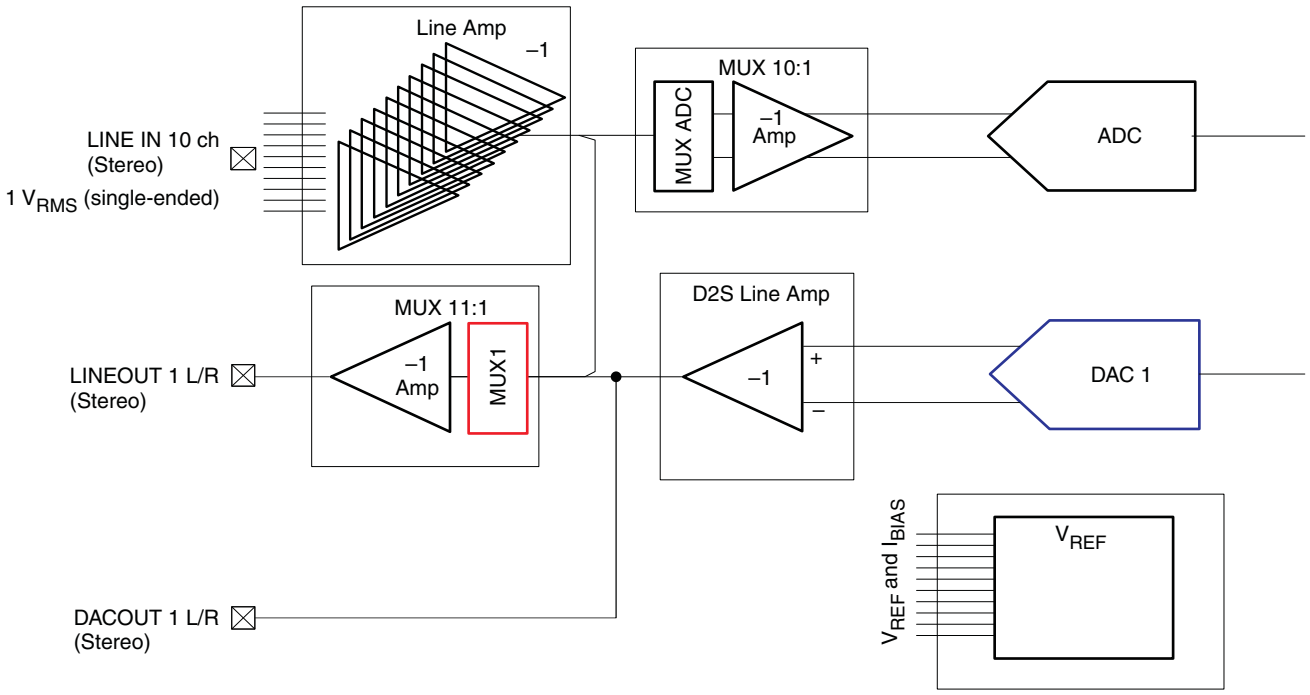
The ADC supports a sampling rate of 48 kHz in clock master mode. In clock slave mode, 32-, 44.1-, and 48-kHz sampling frequencies are supported, based on the master clock frequency.

Stereo Digital-to-Analog Converters (DACs)

The TAS3208 has three stereo DACs. Each DAC can operate a maximum of 48 kHz. The DACs provide a 48-kHz sampling frequency in master mode. In slave mode, 32-, 44.1-, and 48-kHz sampling frequencies are supported, based on the master clock frequency. Two of the DACs are configured for providing line outputs. One of the stereo DACs has the capability to drive either a line out or to be used as a headphone (HP) amplifier.

The stereo HP amplifier is designed to drive up to 24 mW per channel into a headphone speaker load of 16 Ω. The headphone output is a single-ended configuration using series 16-Ω resistors and ac-coupling capacitors.

The TAS3208 includes a multiplexed stereo line driver output. The input can be selected to use the output of the stereo DAC or one of the ten sets of analog inputs. The line driver is capable of driving up to a 10-kΩ load.



Register Map for MUTE Control

0x09

Pin Name	BIT		MUTE Block
LINEOUT1	13	12	MUX1

Pin Name	BIT		MUTE Block
DACOUT1	7	6	DAC 1
DACOUT2	5	4	DAC 2
HPOUT	3	2	DAC 3

		DESCRIPTION
0	0	HW Mute control
*	1	Force MUTE OFF
1	0	Force MUTE ON

Figure 16. Analog Input/Output

Embedded M8051 WARP Microcontroller

The embedded M8051 WARP microcontroller provides the overall control for the TAS3208 device. This control includes device initialization, memory loading, I^2C transactions, control-pin operations, and participation in most processing tasks requiring multiframe processing cycles.

The microcontroller has its own data RAM for storing intermediate values and queuing I^2C commands, a fixed boot program ROM, and a programmable program RAM. The microprocessor's boot program cannot be altered. The microcontroller has specialized hardware for a master and slave interface operation, volume updates, and a programmable interval timer interrupt.

M8051 Addressing Modes

The 256 bytes of internal data memory address space are accessible using indirect addressing instructions (including stack operations). However, only the lower 128 bytes are accessible using direct addressing. The upper 128 bytes of direct address data memory space are used to access external special function data registers (ESFRs).

Register Banks

There are four directly addressable register banks, only one of which may be selected at one time. The register banks occupy Internal data memory addresses from 00 hex to 1F hex.

Bit Addressing

The 16 bytes of internal data memory that occupy addresses from 20 hex to 2F hex are bit addressable. ESFRs that have addresses in the form 1XXXX000 binary are also bit addressable.

Scratch Pad

Internal data memory occupying direct addresses from 30 hex to 7F hex can be used as scratch-pad registers or for the stack.

External Data Memory

External data RAM occupies a 64K address space. This space contains ESFRs. ESFRs permit access and control of the hardware features and internal interfaces of the TAS3208 DSP.

M8051 Boot-Up Sequence

Figure 17 shows the boot-up sequence. M8051 MCU ROM code follows this sequence after device reset release. After the micro completes the boot-up application code (RAM code), the microcontroller switches the program counter from ROM to RAM code by pc_source(esfr - 0xFD).

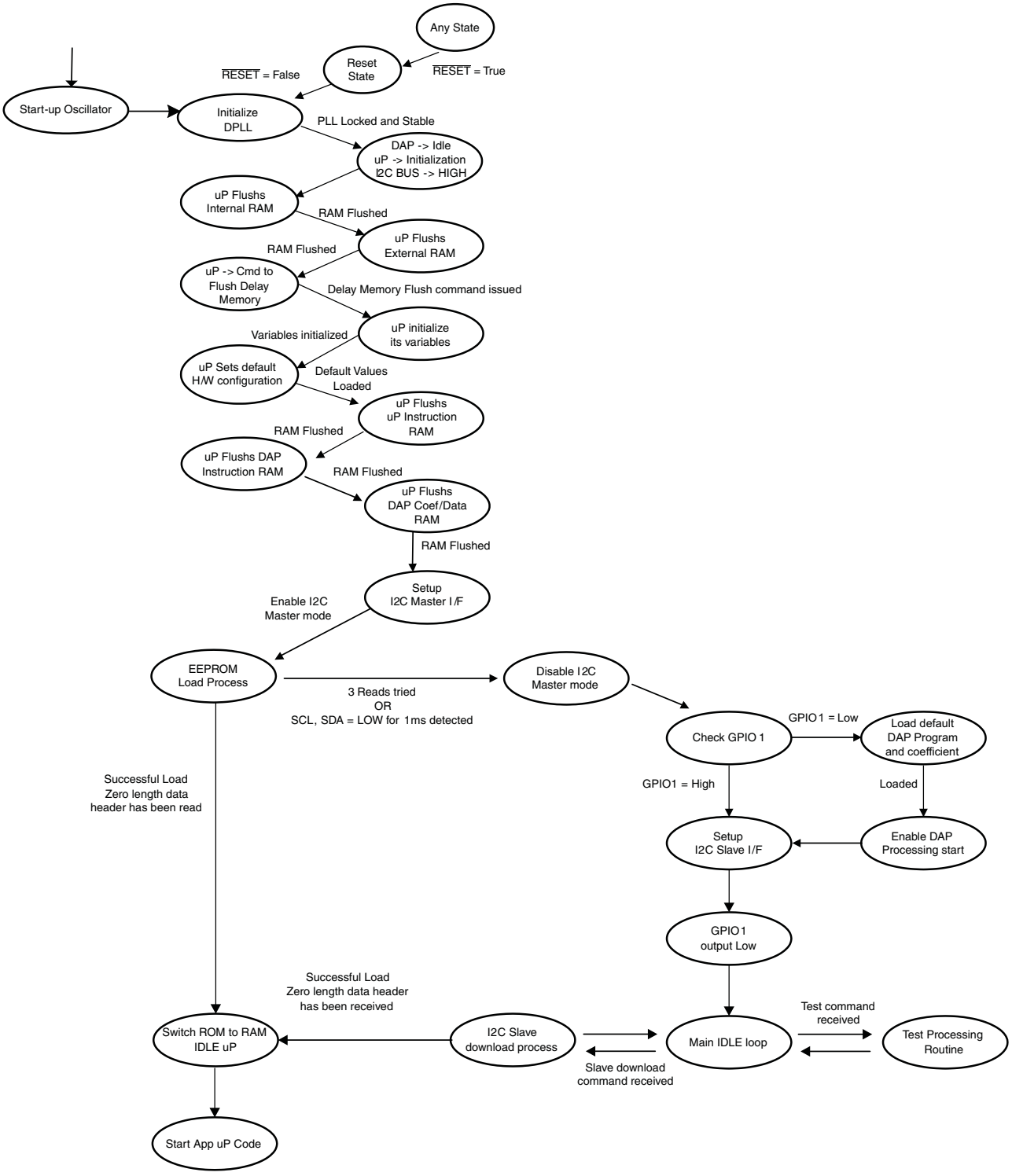


Figure 17. Boot-Up Sequence

Detailed information about the boot-up sequence is given in Table 6.

Table 6. Process Description

PROCESS STATE	ESFR		DESCRIPTION
DSP → idle uP → initialization I²C bus → high			
uP flush internal RAM			Clear micro internal RAM (256 byte)
uP flush external RAM			Clear micro external RAM (2048 byte)
uP command to flush delay memory	clr_dly_ram (0xc0 bit(3))	1	
uP initialize variables			Initialize variables
uP set default H/W configuration	mute0_t	0	Default mutez control
	mute1_t	0	
	mute2_t	0	
	reset_dac_mod	0xff	
	reset_adc_sinc	0x03	
	clock_control1	0x0a	
	clock_delay_control2	0x05	
	clock_delay_sel	0x80	
	i2s_word_byte	0x22	IW/OW: 24 bit IM/OM: I²S
	i2c_mode_byte	0x22	
	sap_en	1	
uP flush uP instruction RAM	mem_sel	0x02	Clear uP instruction RAM (16384 byte)
uP flush DSP instruction RAM	mem_sel	0x01	Clear DSP instruction RAM (3328 W)
uP flush DSP lower coefficient/data RAM	mem_sel	0x00	Clear DSP lower coefficient RAM (1024 W) and data (48 bit) RAM (768 W)
Enable I²C master interface			Setup I²C master interface mode (enable interrupt 10)
EEPROM load			
Disable I²C master mode and enable slave interface	i2c_ms_ctl	0	Switch control MUX to slave I²C port
Switch ROM to RAM	pc_source	1	

Load default DSP Program and coefficient	host_dsp	0	If (gpio_in_3_0 == 1) {
			Host_dsp = 1; /* keep DSP turned off */
			} else {
			Host_dsp = 0; /* turn on DSP */
			}

GPIO1 output low			Enable GPIO output mode, and output low

Control Pins

$\overline{\text{RESET}}$

$\overline{\text{RESET}}$ is an asynchronous control signal that restores all TAS3208 components to the default configuration. When a reset occurs, the digital audio processor (DAP) is put into an idle state and the M8051 MCU starts initialization. A reset can be initiated by inputting logic 0 on the reset pin . A reset will also be issued at power-up sequencing by the internal 1.8-V regulator power subsystem.

NOTE

There is a 1.3-µs deglitch filter on $\overline{\text{RESET}}$.

During a power up sequencing process, $\overline{\text{RESET}}$ should be held low until the DVDD and AVDD power inputs have reached a voltage of 3 V.

As long as $\overline{\text{RESET}}$ is held a logic 0, the device is in the reset state. During this reset state, all I²C and serial data bus operations are ignored. The I²C interface SCL and SDA lines goes HIGH and remain in that state until device initialization has completed.

Power-Up Sequence

The rising edge of the $\overline{\text{RESET}}$ pin begins the initialization of housekeeping functions by clearing memory and setting the default register values. After housekeeping initialization is complete, the TAS3208 enables the master I^2C interface. The TAS3208 then uses the master I^2C interface to determine if an external memory device is present.

External Memory Device Present

Using the master I^2C interface, the TAS3208 will automatically test to see if an external memory device is at address 1010xxx. The value xxx can be chip selects, other information, or don't care depending on the EEPROM selected.

If an external memory device is present and it contains the correct header information along with one or more blocks of program/memory data, the TAS3208 will automatically download the M8051 MCU program RAM, coefficient, and/or data RAM from the external EEPROM. This download is considered complete when an 'end of program' header is read by the TAS3208.

The memory block structure of the external memory device is available in Master I2C Load RAM Block Formats.

At this point, the TAS3208 will disable the master I^2C interface, enable the slave I^2C interface, and start normal operation. After a successful download, the M8051 MCU program counter will be reset and the downloaded M8051 MCU and DSP application firmware will control execution.

External Memory Device Not Present

If no external EEPROM is present or if an error occurred during the external memory device read, the TAS3208 will disable the master I^2C interface, enable the slave I^2C interface. The default slave configuration will then be loaded from the ROM into the M8051 MCU and DSP. In this default configuration, the TAS3208 will stream audio from input to output if the GPIO1 pin is pulled LOW.

NOTE

The master and slave interfaces do not operate simultaneously, thus when one interface is enabled, the other is disabled.

I^2C Chip Select (CS)

The CS pin on the TAS3208 allows up to two TAS3208 devices to be addressed by the I^2C bus via an external host controller, without the need for external logic. Table 7 and Table 8 list the I^2C address for each I^2C interface.

Table 7.
I^2C Slave Addressing

SLAVE ADDRESS	CS
0x68/69	0
0x6A/6B	1

Table 8.
I^2C Master Addressing

SLAVE ADDRESS	CS
0xA0/A1	0
0xA2/A3	1

General-Purpose Input/Output (GPIO) Pins

The TAS3208 has two level-sensitive GPIO pins, GPIO1 and GPIO2, that are firmware programmable. Upon power up or following a $\overline{\text{RESET}}$, GPIO1 becomes an input and has a special function as described in GPIO1 Pin Function.

GPIO1 Pin Function

- After $\overline{\text{RESET}}$ or power-up initialization, if no EEPROM is present, a memory error occurs, or SDA and SCL are pulled LOW for 1 ms, then TAS3208 will disable the master I^2C interface and enable the slave I^2C interface initialization, to load the slave default configuration.
  - When GPIO1 has been pulled HIGH through a 10-kΩ to 20-kΩ resistor, the TAS3208 will then initialize in the default configuration with the serial data outputs not active. Once the TAS3208 has completed its default initialization procedure, with the Status register updated and the I^2C slave interface enabled, GPIO1 will become an output and will be driven LOW. Following the HIGH to LOW transition of GPIO1, the system controller can access the TAS3208 through the I^2C interface and read the Status register to determine the load status.

    If a memory read error occurs, the TAS3208 reports the error in the Status register.
  - When GPIO1 has been pulled LOW through a 10-kΩ to 20-kΩ resistor to permit a simple functional device test, GPIO1 can be pulled LOW using external logic and a 10-kΩ to 20-kΩ resistor. In this case, once the TAS3208 has completed its default test initialization procedure, with the Status register updated and the I^2C slave interface enabled, the TAS3208 will stream audio from the input SDIN1 to outputs SDOUT1 and SDOUT2.

    At this point, GPIO1 becomes an output and will be driven LOW. If the external logic is no longer driving GPIO1 LOW after the load has completed (≠100 ms following a RESET if no EEPROM is present), the state of GPIO1 can be observed. At this point, the system controller can access the TAS3208 through the I^2C interface and read the Status register to determine the load status.

NOTE

If the GPIO1 pin state is not observed, the only indication that the device has completed its initialization procedure is that the TAS3208 will stream audio and the I^2C slave interface has been enabled.

NOTE

Some I^2C masters will hang when they receive a NAC during an I^2C transaction.

- Once the TAS3208 has been programmed either through a successful boot load or via slave I^2C download, the operation of GPIO1 can be programmed to be an input or an output.

GPIO Ports

In I^2C slave mode, the GPIO ports can be used as true general-purpose ports. Each port can be individually programmed via the I^2C bus to be either an input or output port. The default assignment for all GPIO ports in I^2C slave mode is an input port.

When a given GPIO port is programmed as an output port, by setting the appropriate bit in the bit field GPIODIR of subaddress 0x0C to logic 1, the logic-level output is set by the logic level programmed into the appropriate bit in bit field GPIO IN OUT. The I^2C bus then controls the logic output level for those GPIO ports assigned as output ports. When a given GPIO port is programmed as an input port by setting the appropriate bit in bit field GPIODIR to logic 0, the logic input level into the GPIO port is written to the appropriate bit in bit field GPIO IN OUT. The I^2C bus then can be used to read bit field GPIO IN OUT to determine the logic levels at the input GPIO ports. Whether a given bit in the bit field GPIO IN OUT is a bit to be read via the I^2C bus or a bit to be written to via the I^2C bus is strictly determined by the corresponding bit setting in bit field GPIODIR.

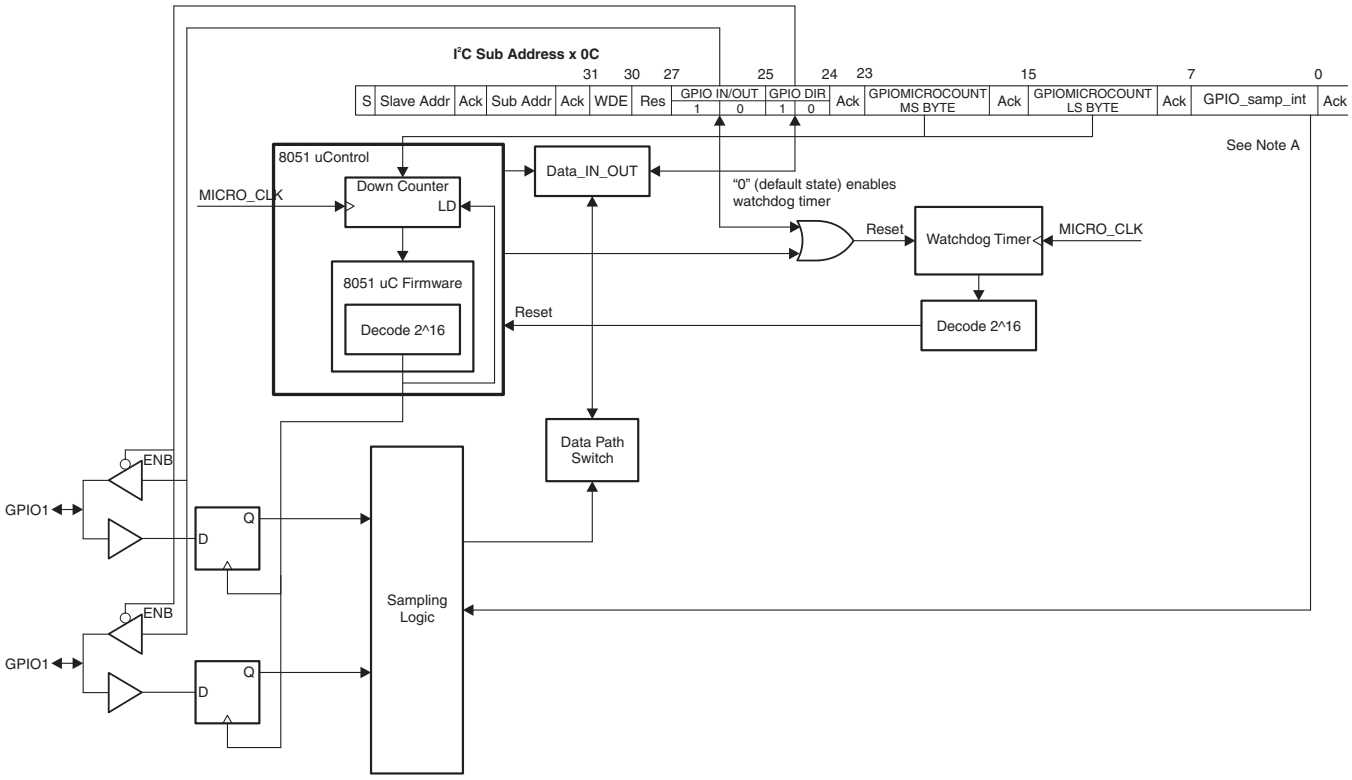
In I^2C slave mode, the GPIO input ports are read every GPIOMICROCOUNT micro clocks, as was the case in the I^2C master mode. However, parameter GPIO_samp_int does not have a role in I^2C slave mode. If a GPIO port is assigned as an output port, a logic 0 bit value is supplied by the TAS3208 for this GPIO port in response to a read transaction at subaddress 0x0C.

If the GPIO ports are left in their power turnon default state, they are input ports with a weak pullup on the input to VDSS.

Watchdog Timer

There is a hardware watchdog timer in the TAS3208 that can be programmed in the customer application code to monitor the microprocessor activity. If the watchdog timer expires, it will generate a reset to the 8051 microprocessor. GPIOMICROCOUNT, in subaddress 0x0C, is used in order to trigger GPIO and the monitoring to the DSP diagnostic count. Because of this, the value selected for GPIOMICROCOUNT must be chosen to provide a good tradeoff of between micro overheard and adequate execution frequency of these processes. The default value for this counter is 0x5820 which corresponds to a period of 1.25 ms.

Figure 18 shows the GPIO register, the GPOI interface, and a typical user application code implementation of the watchdog timer reset.



A. Determines how many consecutive logic 0 samples (where each sample is spaced by GPIOMICROCOUNT Micro_clks) are required to read a logic 0 on a GPIO input port

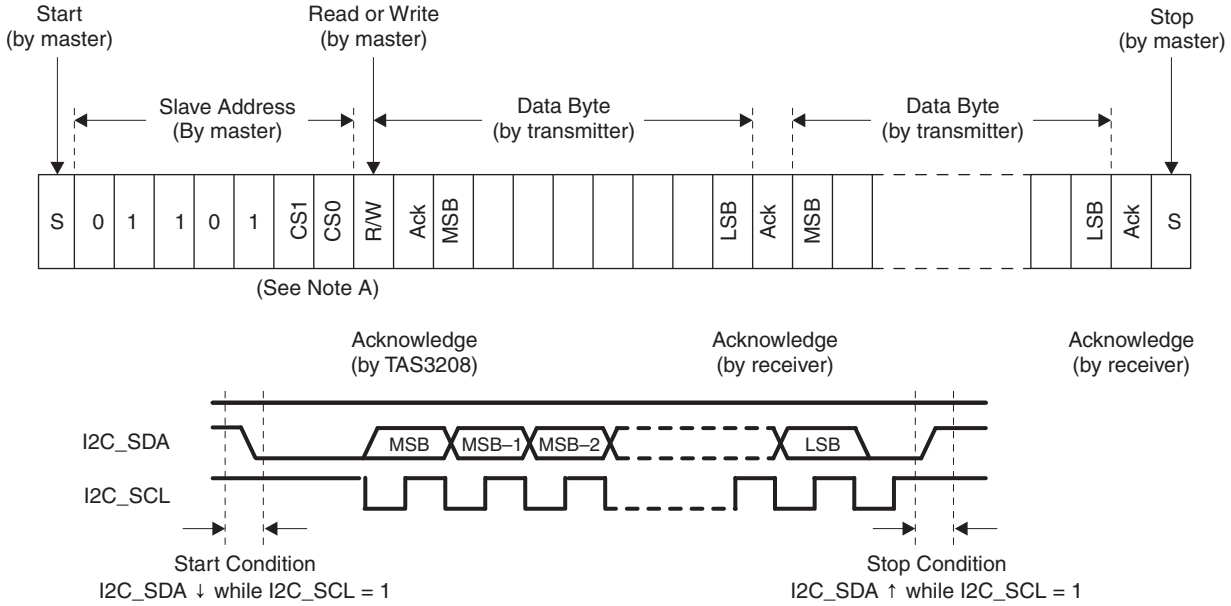
Figure 18. GPIO Ports

I²C Control Interface

General I²C Transactions

The M8051 microprocessor receives and distributes I²C data to the I²C bus controllers, and participates in most I²C processing tasks requiring multiframe processing cycles. The master and slave interfaces do not operate simultaneously.

The I²C communication protocol for the I²C slave mode is shown in Figure 19.



A. Bits CS1 and CS0 in the TAS3208 slave address are compared to the logic levels on pins CS0 and CS1 for address verification. This provides the ability to address up to four TAS3208 chips on the same I²C bus.

Figure 19. I²C Slave-Mode Communication Protocol

The I²C bus employs two signals – SDA (data) and SCL (clock) – to communicate between integrated circuits in a system. Data is transferred on the bus serially one bit at a time. The address and data be transferred in byte (8-bit) format with the MSB transferred first. In addition, each byte transferred on the bus is acknowledged by the receiving device with an acknowledge bit. Each transfer operation begins with the master device driving a Start condition on the bus and ends with the master device driving a Stop condition on the bus. The bus uses transitions on the data (SDA) terminal while the clock is HIGH to indicate Start and Stop conditions. A HIGH-to-LOW transition on SDA indicates a Start, and a LOW-to-HIGH transition indicates a Stop. Normal data bit transitions must occur within the low time of the clock period. The master generates the 7-bit slave address and the read/write (R/W) bit to open communication with another device and then waits for an acknowledge condition. The slave holds SDA LOW during the acknowledge clock period to indicate an acknowledgement. When this occurs, the master transmits the next byte of the sequence. Each device is addressed by a unique 7-bit slave address plus R/W bit (1 byte). All compatible devices share the same signals via a bidirectional bus using a wired-AND connection. An external pullup resistor must be used for the SDA and SCL signals to set the HIGH level for the bus.

There is no limit on the number of bytes that can be transmitted between Start and Stop conditions. When the last word transfers, the master generates a Stop condition to release the bus.

A read transaction requires that the master device first issue a write transaction to give the TAS3208 the subaddress to be used in the read transaction that follows. This subaddress assignment write transaction is then followed by the read transaction. For write transactions, the subaddress is supplied in the first byte of data written, and this byte is followed by the data to be written. For write transactions, the subaddress must always be included in the data written. There cannot be a separate write transaction to supply the subaddress, as was required for read transactions. If a subaddress assignment's only write transaction is followed by a second write transaction supplying the data, erroneous behavior results. The first byte in the second write transaction is interpreted by the TAS3208 as another subaddress replacing the one previously written.

Multiple Byte Write

A multiple byte data write transfer is identical to a single byte data write transfer, except that multiple data bytes are transmitted by the master device to slave (see Figure 20). After receiving each data byte, the TAS3208 will respond with an acknowledge bit.

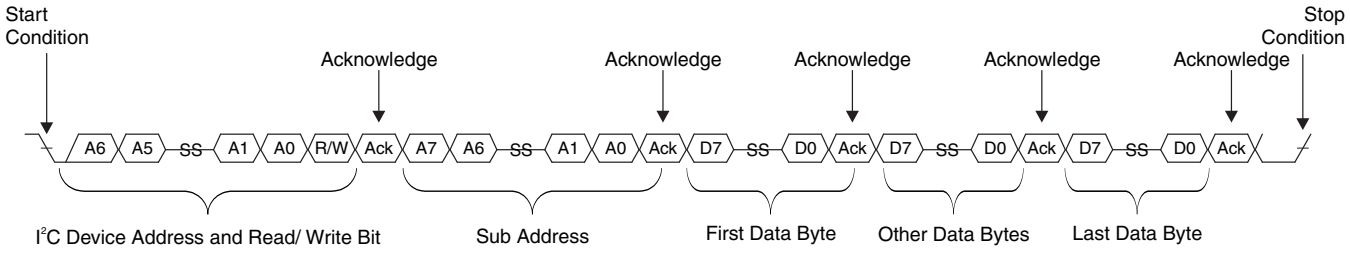


Figure 20. Multiple Byte Write Transfer

Multiple Byte Read

A multiple byte data read transfer is identical to a single byte data read transfer, except that multiple data bytes are transmitted by the TAS3208 to the master device (see Figure 21). Except for the last data byte, the master device will respond with an acknowledge bit after receiving each data byte.

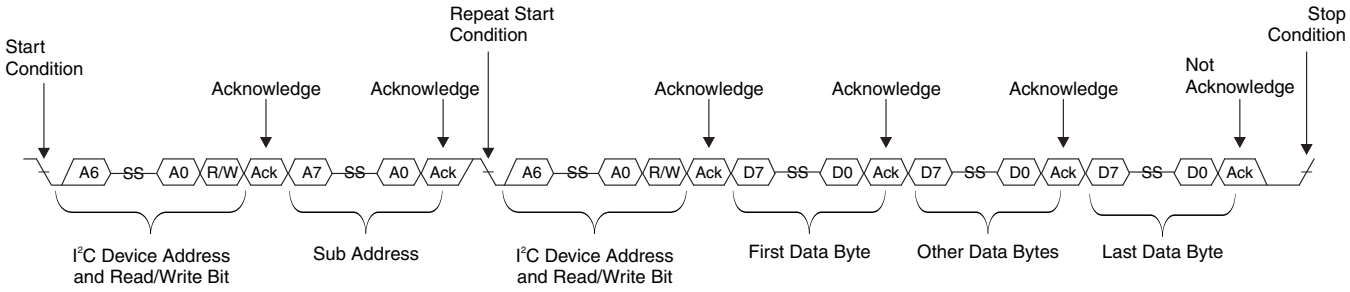


Figure 21. Multiple Byte Read Transfer

Random I²C Transactions

Supplying a subaddress for each subaddress transaction is referred to as random I^2C addressing. For random I^2C read commands, the TAS3208 responds with data, a byte at a time, starting at the subaddress assigned, as long as the master device continues to respond with acknowledges. If a given subaddress does not use all 32 bits, the unused bits are read as logic 0. I^2C write commands, however, are treated in accordance with the data assignment for that address space. For example, if a write command is received for a biquad subaddress, the TAS3208 expects to see five 32-bit words. If fewer than five data words have been received when a Stop command (or another Start command) is received, the data received is discarded.

Sequential I²C Transactions

The TAS3208 supports sequential I^2C addressing. For write transactions, if a subaddress is issued followed by data for that subaddress and the 15 subaddresses that follow, a sequential I^2C write transaction has taken place, and the data for all 16 subaddresses is successfully received by the TAS3208. For I^2C sequential write transactions, the subaddress then serves as the start address and the amount of data subsequently transmitted, before a Stop or Start is transmitted, determines how many subaddresses are written to. As was true for random addressing, sequential addressing requires that a complete set of data be transmitted. If only a partial set of data is written to the last subaddress, the data for the last subaddress is discarded. However, all other data written is accepted; just the incomplete data is discarded.

Sequential read transactions do not have restrictions on outputting only complete subaddress data sets.

If the master does not issue enough data received acknowledges to receive all the data for a given subaddress, the master device simply does not receive all the data.

If the master device issues more data received acknowledges than required to receive the data for a given subaddress, the master device simply receives complete or partial sets of data, depending on how many data received acknowledges are issued from the subaddress(es) that follow. I²C read transactions, both sequential and random, can impose wait states.

For the standard I²C mode (SCL = 100 kHz), worse-case wait state times for an 8-MHz microprocessor clock is on the order of 2 µs. Nominal wait state times for the same 8-MHz microprocessor clock is on the order of 1 µs. For the fast I²C mode (SCL = 400 kHz) and the same 8-MHz microprocessor clock, worse-case wait state times can extend up to 10.5 µs in duration. Nominal wait state times for this same case lie in a range from 2 µs to 4.6 µs. Increasing the microprocessor clock frequency lowers the wait state times and for the standard I²C mode, a higher microprocessor clock can totally eliminate the presence of wait states.

For example, increasing the microprocessor clock to 16 MHz results in no wait states. For the fast I²C mode, higher microprocessor clocks shortens the wait state times encountered, but does not totally eliminate their presence.

I²C Master-Mode Operation

I²C master-mode operation is enabled following a reset or power-on reset.

The TAS3208 uses the master mode to download from EEPROM the memory contents for:

- Microprogram memory
- Micro extended memory
- DSP program memory
- DSP coefficient memory
- DSP data memory

The TAS3208, when operating as an I²C master, can execute a complete download of any internal memory or any section of any internal memory without requiring any wait states.

When the TAS3208 operates as an I²C master, it generates a repeated Start without an intervening Stop command while downloading program and memory DATA from an external EEPROM. When a repeated Start is sent to the EEPROM in read mode, the EEPROM enters a sequential read mode to quickly transfer large blocks of data.

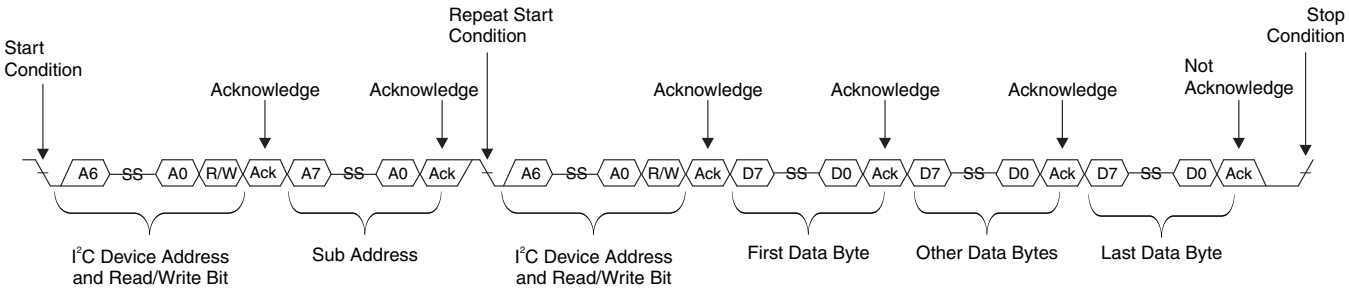


Figure 22. Multiple-Byte Read Transfer

The TAS3208 will query the bus for an I²C EEPROM at an address 1010xxx. The value xxx can be chip selects, other information, or don't cares depending on the EEPROM selected.

The first act of the TAS3208 as master will be to transmit a Start condition along with the device address of the I²C EEPROM, with the read/write bit cleared (0) to indicate a write. The EEPROM acknowledges the address byte and the TAS3208 sends a subaddress byte, which the EEPROM will acknowledge. Most EEPROMs have at least 2-byte addresses and will acknowledge as many as are appropriate. At this point, the EEPROM sends a last acknowledge and becomes a slave transmitter. The TAS3208 acknowledges each byte repeatedly to continue reading each data byte that is stored in memory.

The memory load information starts with reading the header and data information that starts at subaddress 0 of the EEPROM. This information must be stored in a sequential memory addresses with no intervening gaps. The data block is contiguous blocks of data that immediately follow the headers' locations. The TAS3208 memory data can be stored and loaded in (almost) any order. Additionally this addressing scheme permits portions of the TAS3208 internal memories to be loaded.

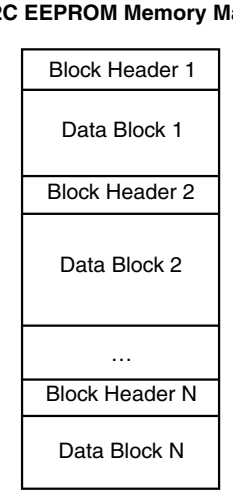


Figure 23. EEPROM Address Map

The TAS3208 will sequentially read EEPROM memory and load its internal memory unless it does not find a valid memory header block, is not able to read the next memory location because the end of memory was reached, detects a checksum error, or reads a end-of-program header block. When it encounters a valid header or read error, the TAS3208 will attempt to read the header or memory location three times before it determines that it has an error. If the TAS3208 encounters a checksum error, it will attempt to reread the entire block of memory two more times before it determines that it has an error.

NOTE

Once the microprogram memory has been loaded, it can not be reloaded until the TAS3208 has been reset.

If an error is encountered, the TAS3208 terminates its memory load operation, loads the default configuration for both the M8051 MCU and DSP from the embedded ROM, and disables further master I^2C bus operations.

If an end-of-program data block is read, the TAS3208 has completed the initial program load.

The I^2C master mode utilizes the starting and ending I^2C checksums to verify a proper EEPROM download. The first 16-bit data word received from the EEPROM is the I^2C checksum at subaddress 0x00. It is stored and compared against the 16-bit data word received for last subaddress, the ending I^2C checksum, and the checksum that is computed during the download. These three values must be equal. If the read and computed values do not match, the TAS3208 sets the memory read error bits in the Status register and repeats the download from the EEPROM two more times. If the comparison check again fails the third time, the TAS3208 sets the microprogram to the default value.

NOTE

When acting as an I^2C master, the data rate transfer is fixed at 375 kHz.

I²C Slave Mode Operation

The I²C slave mode is the mode that is used to change configuration parameters during operation and perform program and coefficient downloads from a master device. The latter can be used to replace the I²C master mode EEPROM download.

The TAS3208 uses the slave mode to load the memory contents for the:

- Microprogram memory
- Micro extended memory
- DSP program memory
- DSP coefficient memory
- DSP data memory
- Update coefficient and other control values
- Read status flags

The TAS3208 support both random and sequential I²C transactions. The TAS3208 I²C slave address is 011010X, where the first six bits are the TAS3208 device address and the final one bit is set by the TAS3208 internal microprocessor at power up. The internal microprocessor derives the last bit from an external pin (CS),which is pulled up or down to create two unique addresses for control of multiple TAS3208 part applications. The pulldown resistance of CS creates a default 00 address when no connection is made to the pin.

The TAS3208 I²C block does respond to the broadcast address (00h).

NOTE

When acting as an I²C slave, data-rate transfer is determined by the master device on the bus. However, the setting of I²C parameter N at subaddress 0x01 does play a role in setting the maximum possible data transfer rate. In the I²C slave mode, bit rates other than (and including) the I²C-specific 100-Kbps and 400-Kbps bit rates can be obtained, but N must always be set so that the oversample clock into the I²C master and slave controllers is at least a factor of 20 higher in frequency than SCL.

N = 0 is a special case. When N = 0, a mode is enabled that detects I²C frames and enables the TAS3208 I²C interface to reset and continue operation after receiving an invalid I²C frame.

Table 9.
I²C Slave Addresses

SLAVE ADDRESS	CS
0x68/69	0
0x6A/6B	1

Table 10.
I²C Master Addresses

SLAVE ADDRESS	CS
0xA0/A1	0
0xA2/A3	1

Digital Signal Processor (DSP) Arithmetic Unit

Overview

The arithmetic processor is a fixed-point computational engine consisting of an arithmetic unit and data and coefficient memory blocks. The primary features are:

- Two pipe parallel processing architecture
  - 48-bit data path with 76-bit accumulator
  - Hardware single-cycle multiplier (28 × 48)
  - Three 48-bit general-purpose data registers
  - One 28-bit coefficient register
  - 48-bit adder
  - 28-bit adder
  - Shift right, shift left
  - Bimodal clip
  - Log2/Alog2
  - Magnitude truncation
- Read/read/write single-cycle memory access
- Data input is 48-bit 2s complement multiplexed in from SAP immediately following FSYNC pulse.
- Data output is four 32-bit 2s-complement buses.
- Separate control for writing to delay memory
- Separate coefficient memory (28 bit) and data memory (48 bit)
- Linear Feedback Shift Register (LFSR) in the instruction register doubles as a random number generator in normal operating mode.
- Coefficient RAM, data RAM, LFSR seed, program counter, and memory pointers are all mapped into the same memory space for convenient addressing by the micro.
- Memory interface block contains four pointers – two for data memory and two for coefficient memory.

Data Format

Figure 24 shows the data word structure of the arithmetic unit. Eight bits of overhead or guard bits are provided at the upper end of the 48-bit word, and 16 bits of computational precision or noise bits are provided at the lower end of the 48-bit word. The incoming digital audio words are all positioned with the MSB abutting the 8-bit overhead/guard boundary. The sign bit in bit 39 indicates that all incoming audio samples are treated as signed data samples.

The arithmetic engine is a 48-bit (25.23 format) processor consisting of a general-purpose 76-bit arithmetic logic unit and function-specific arithmetic blocks. Multiply operations (excluding the function-specific arithmetic blocks) always involve 48-bit words and 28-bit coefficients (usually I^2C programmable coefficients). If a group of products are to be added together, the 76-bit product of each multiplication is applied to a 76-bit adder, where a DSP-like multiply-accumulate (MAC) operation takes place. Biquad filter computations use the MAC operation to maintain precision in the intermediate computational stages.

To maximize the linear range of the 76-bit ALU, saturation logic is not used. In MAC computations, intermediate overflows are permitted, and it is assumed that subsequent terms in the computation flow will correct the overflow condition.

The memory banks include a dual-port data RAM for storing intermediate results, a coefficient RAM, and a fixed-program ROM. Only the coefficient RAM, assessable via the I^2C bus, is available to the user.

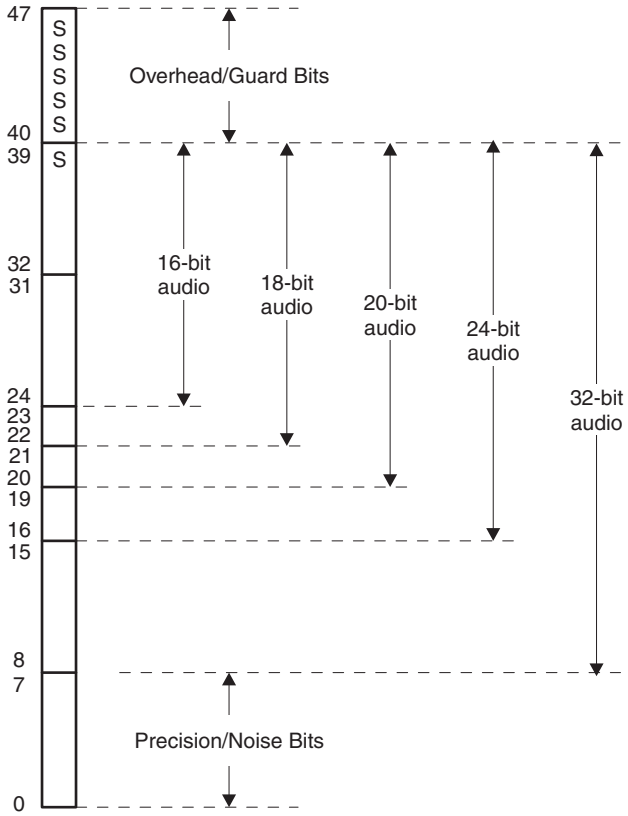


Figure 24. Arithmetic Unit Data Word Structure

8-Bit ALU Operation
(without saturation)

	10110111	(–73)	–73
+	11001101	(–51)	+ –51
	10000100	(–124)	–124
+	11010011	(–45)	+ –45
Rollover →	01010111	(57)	–169
+	00111011	(59)	+ 59
	10010010	(–110)	–110

Figure 25. DSP ALU Operation With Intermediate Overflow

DAP Data Path Data Representation

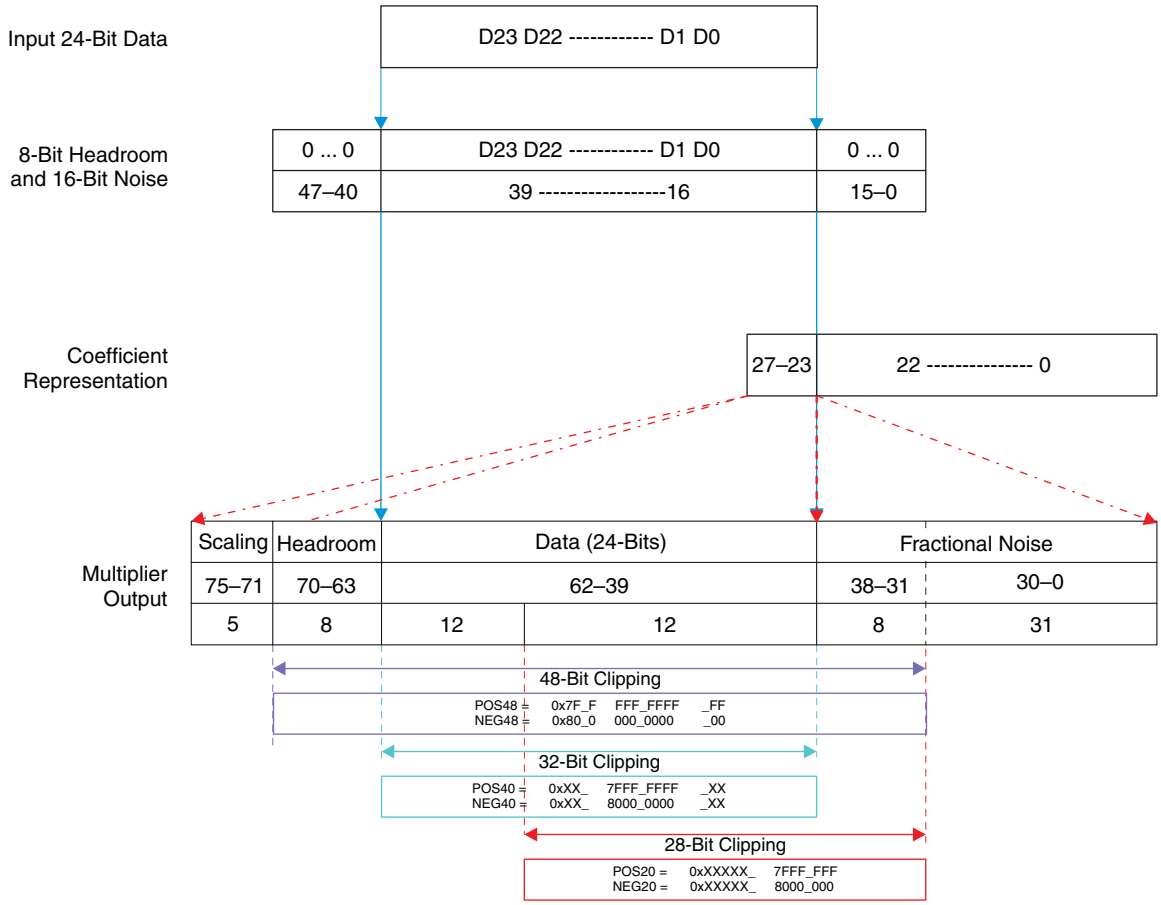


Figure 26. DSP Data-Path Data Representation

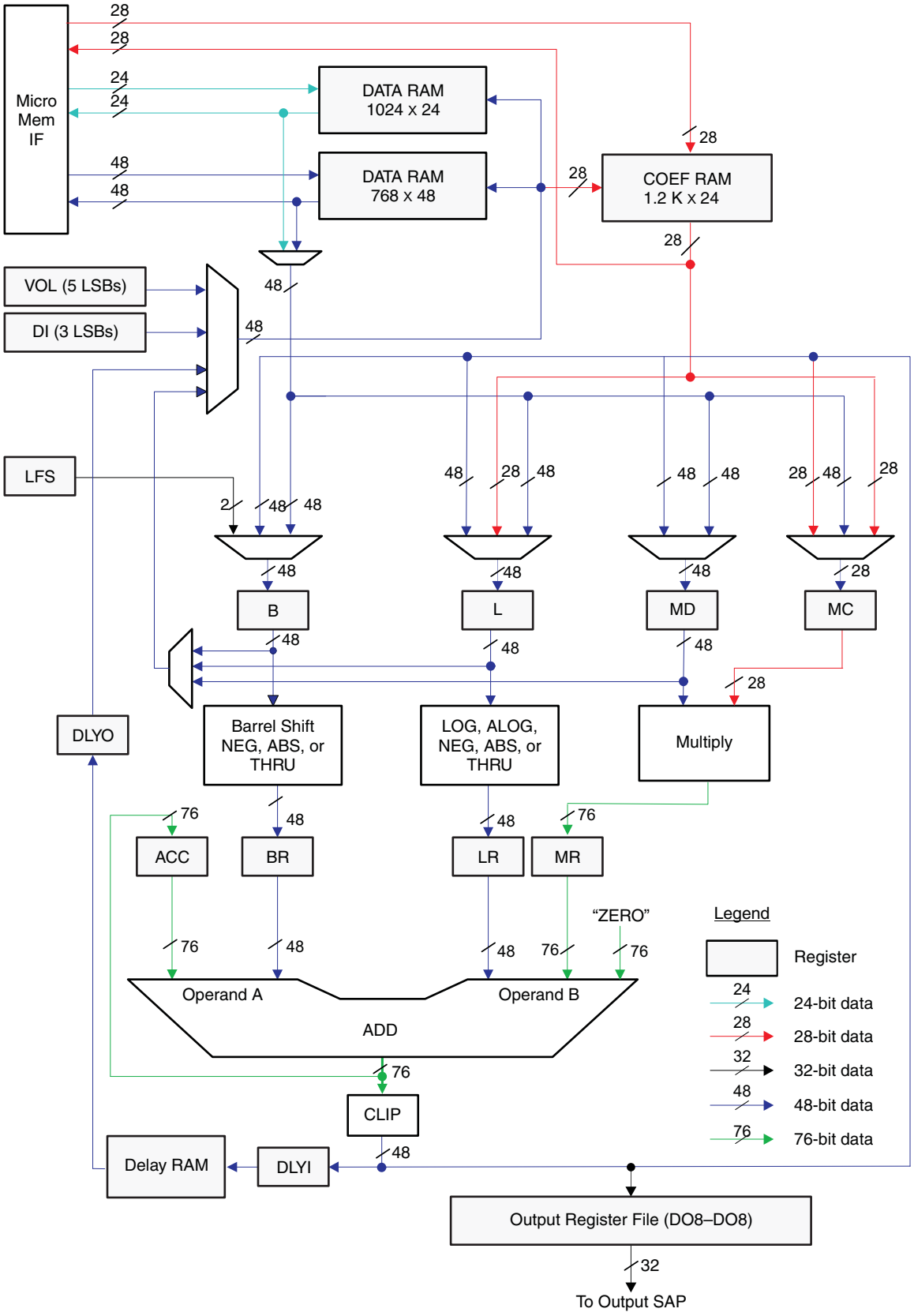


Figure 27. DSP Data-Path Architecture

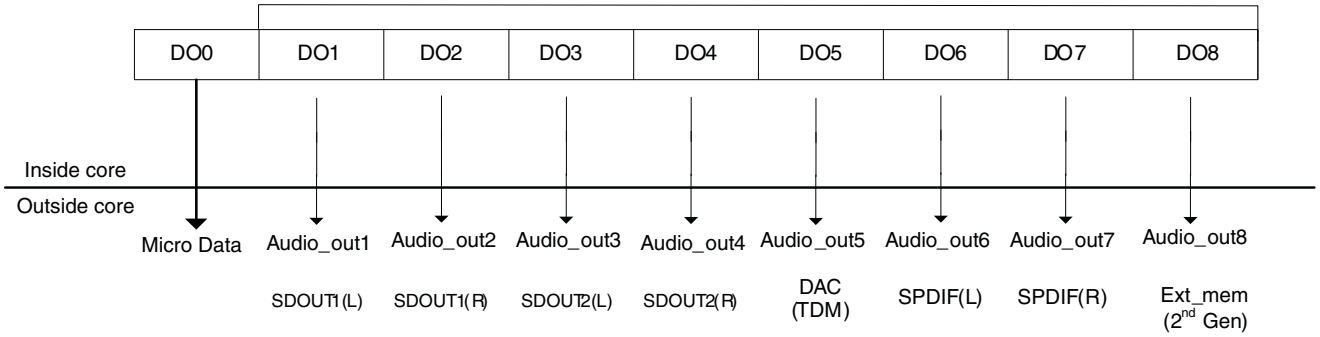
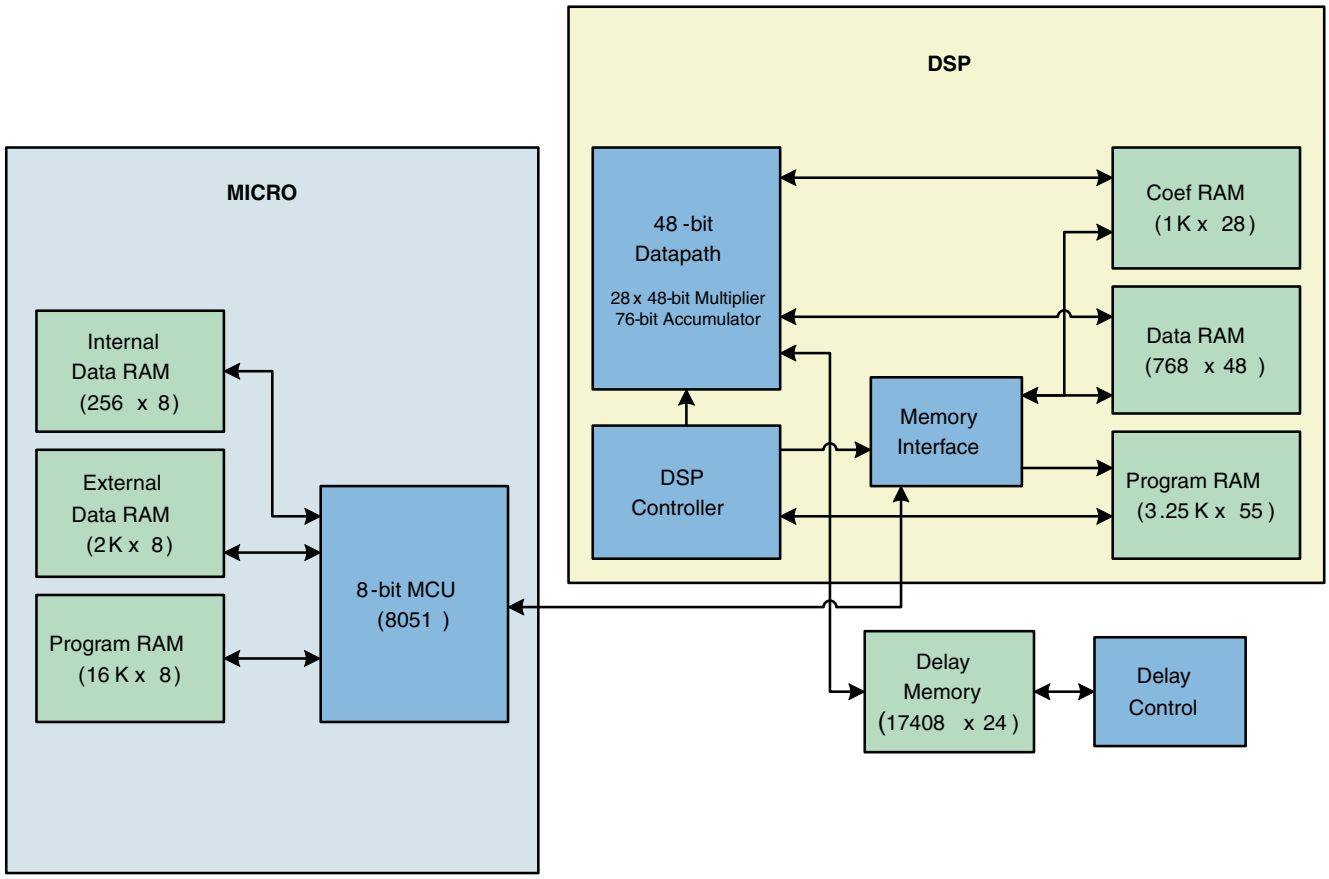


Figure 28. DSP Output Register Configuration



A. Memory size K = 1024

Figure 29. DSP, MCU, and Memory Interfaces

Delay Memory

The delay memory interface (DMIF) is the interface block between the DSP core and the delay memory. The DMIF block's primary purpose is to keep track of 24 sets of delay memory pointers that are initially set up by the microcontroller through an I^2C command(s). Eight of the pointers are used to write/retrieve 48-bit data (full-precision intermediate) and the other 16 for 24-bit data (post quantized). Thus, to support 48-bit word reverb delay, two RAM locations must be used.

The key features of the delay memory are:

- 17408 × 24 delay memory locations
- 24 separately addressable pointers
- Programmable Start/Stop address on each pointer
- Pointers capable of accessing 24-bit or 48-bit words
- Single-port access (one pointer access per access cycle)
- Access cycle < four DSP clocks
- Self clearing – INIT pin used to clear all memory to zero
- Fully synchronous
- DP1–DP15: 16 24-bit pointers
- RP1–RP8: Eight 48-bit (full precision) pointers

Since all of the pointers are contiguous, it is only necessary to write the address END point. For example, if DP1 is to be a three-sample delay, the register DP1 should be set to 0x003. If RP1 is to be a three-sample delay, the register RP1 should be set to the value of DP15 + 6. All of the DP16–DP1 and RP8–RP1 registers must be set to a minimum of a one sample delay (one or two words).

DP1 Start address is defined as 000x0.

DP2 Start address is equal to DP1 end address + 1.

RP1 Start address is equal to DP16 end address + 1.

RP8 Start address is equal to RP7 end address + 2.

Since the Start/Stop address for each pointer is programmable anywhere in the delay RAM's address space, the delay for any one channel can be anywhere in the delay RAM. There is, however, no address space collision avoidance logic to separate the pointers. The user (or micro) must take care to avoid overlapping the address spacing of each pointer.

Pointer register address endpoint registers DP16–DP1 and RP8–RP1 are typically written only during the initialization (fast load) mode of the device. Writing to these registers while the TAS3208 DSP core is accessing the pointers may cause the pointers to cross the address space of another pointer.

To write to the delay RAM, the TAS3208 DSP core controller must present the data to be written on the PT_DATA bus (LSB always in bit zero of the bus), select the pointer to be accessed by driving the PT_SEL pins, and assert the PT_WZ pin for a minimum of four clocks. The pointer will not increment until a write has been performed and the PT_WZ pin has been deasserted.

To perform a read, the PT_OUT bus may be read four clocks after PT_SEL is driven.

DSP Instruction Word

TAS3208 has a 55-bit instruction word. Each instruction has five independent operations, which can load two operands from data memory and coefficient memory, store the result into data or coefficient memory, and perform two parallel arithmetic operations.

55-BIT INSTRUCTION								
Ext	**ALU First Stage**	**ALU Second Stage**	**Data Memory Load**		**Coefficient Memory Load**		**Memory Store**	
0	P1OP	P2OP	MOP1	AD1	MOP2	AD2	MOP3	AD3
54	53–49	48–42	41–37	36–27	26–24	23–14	13–10	9–0

Figure 30. Instruction Word

The TAS3208 instruction set is a superset of the TAS3208 instruction set, extending the DSP processing capabilities for improved efficiency of FIR operations, as well as extending the addressable memory space. The Ext instruction bit (bit 54) has been added to extend the internal memory address space by one bit, increasing the memory space from 1K to 2K words.

The superset instruction word maintains backward compatibility with the 54-bit instruction word of the TAS3208 device, since the 54-bit instruction word required dummy storage of two bits in the EEPROM.

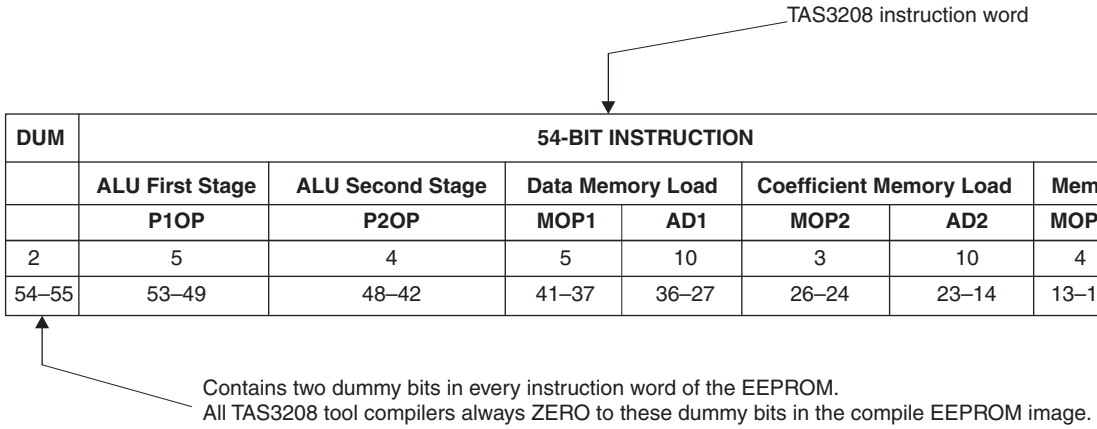


Figure 31. Instruction Word

As shown in Figure 32, the extension bit designates an offset of 1K to all three addresses in the instruction word. However, it should be noted that both data and coefficient memory addresses above the 1K boundary are reserved for housekeeping processing tasks. Any attempt to write to these addresses may corrupt the audio output.

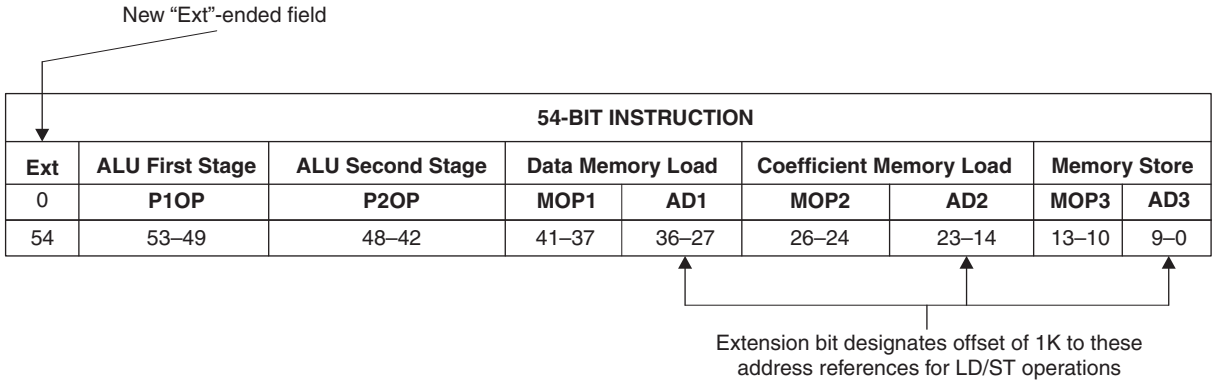


Figure 32. Instruction Word Extension Field

DSP Instruction Set

Please see the *TASxxx Programmer's Guide* for detailed information regarding programming of this device.

ABSOLUTE MAXIMUM RATINGS(1)

			MIN	MAX	UNIT
DVDD	Supply voltage range		–0.5	3.8	V
AVDD	Supply voltage range		–0.5	3.8	V
V_I	Input voltage range	3.3-V TTL	–0.5	VDDS + 0.5	V
		3.3-V analog	–0.5	AVDDS + 0.5	
		1.8-V LVCMOS	–0.5	AVDD(2) + 0.5	
V_O	Output voltage range	3.3-V TTL	–0.5	VDDS + 0.5	V
		3.3-V analog	–0.5	AVDDS + 0.5	
		1.8-V LVCMOS	–0.5	DVDD(3) + 0.5	
			–0.5	AVDD(4) + 0.5	
I_{IK}	Input clamp current	V_I < 0 or V_I > DVDD)		±20	mA
I_{OK}	Output clamp current	V_O < 0 or V_O > DVDD		±20	mA
T_{stg}	Storage temperature range		–65	150	°C
	Lead temperature 1.6 mm (1/16 in) from case for 10 s			260	°C

(1) Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under Recommended Operating Conditions is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) AVDD is an internal 1.8-V supply derived from a regulator in the TAS3208 chip. Pin XTALI is the only TAS3208 input that is referenced to this 1.8-V logic supply. The absolute maximum rating listed is for reference; only a crystal should be connected to XTALI.
(3) DVDD is an internal 1.8-V supply derived from regulators in the TAS3208 chip. DVDD is routed to DVDD_BYPASS_CAP to provide access to external filter capacitors, but should not be used to source power to external devices.
(4) Pin XTALO is the only TAS3208 output that is derived from the internal 1.8-V logic supply AVDD. The absolute maximum rating listed is for reference; only a crystal should be connected to XTALO. AVDD is also routed to AVDD_BYPASS_CAP to provide access to external filter capacitors, but should not be used to source power to external devices.

PACKAGE DISSIPATION RATINGS(1) (2)

PACKAGE	T_A ≤ 25°C POWER RATING	DERATING FACTOR ABOVE T_A = 25°C	T_A = 70°C POWER RATING
TQFP – PZP	2.78 W	28.7°C/W	1.22 W

(1) High-K Board, 105°C junction
(2) Refer to the application report *PowerPAD ™ Thermally Enhanced Package* (literature number SLMA002)

RECOMMENDED OPERATING CONDITIONS

			MIN	NOM	MAX	UNIT
DVDD	Digital supply voltage		3	3.3	3.6	V
AVDD	Analog supply voltage	3.3-V analog	3	3.3	3.6	V
V_{IH}	High-level input voltage	3.3-V TTL	2			V
		1.8-V LVCMOS (XTL_IN)	1.26		1.95	
V_{IL}	Low-level input voltage	3.3-V TTL			0.8	V
		1.8-V LVCMOS (XTL_IN)			0.54	
T_A	Operating ambient air temperature (ensuring parametric)		–20	25	70	°C
T_J	Operating junction temperature		–20		105	°C

AUDIO SPECIFICATIONS – CHANNEL (INPUT TO OUTPUT)

T_A = 25°C, AVDD = 3.3 V, DVDD = 3.3 V, Fs (audio) = 48 kHz, clock source from XTALI, AES17 filter, second-order 30-kHz low-pass filter (unless otherwise noted)

PARAMETER		CONDITIONS	MIN	TYP	MAX	UNIT
Overall dynamic range	A-in → ADC → DSP → DAC → Lineout	A: WTD	87	92		dB
	A-in → MUX → Lineout	A-WTD	95	98		

AUDIO SPECIFICATIONS – DIGITAL FILTERS

T_A = 25°C, AVDD = 3.3 V, DVDD = 3.3 V, Fs (audio) = 48 kHz, clock source from XTALI, AES17 filter, second-order 30-kHz low-pass filter (unless otherwise noted)

PARAMETER	MIN	TYP	MAX	UNIT
ADC Decimation Filter, Fs = 48 kHz				
Filter gain from 0 Fs to 0.39 Fs		±0.1		dB
Filter gain at 0.4125 Fs		–0.25		dB
Filter gain at 0.45 Fs		–3		dB
Filter gain at 0.5 Fs		–17.5		dB
Filter gain from 0.55 Fs to 64 Fs		–75		dB
Filter group delay		17/Fs		s
DAC Interpolation Filter, Fs = 48 kHz				
Pass band	20		0.45 × Fs	Hz
Pass-band ripple		±0.06		dB
Transition band	0.45 × Fs		0.5501 × Fs	Hz
Stop band	0.5501 × Fs		7.455 × Fs	kHz
Stop-band attenuation		–65		dB
Filter group delay		21/Fs		s

ELECTRICAL SPECIFICATIONS – ANALOG SECTIONS(1)

T_A = 25°C, AVDD = 3.3 V, DVDD = 3.3 V, Fs (audio) = 48 kHz, clock source from XTALI, AES17 filter, second-order 30-kHz low-pass filter (unless otherwise noted)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
Stereo MUX Input/ADC Channel	1-kHz sine-wave input				
Full-scale input voltage (0 dB)			1	1.15	Vrms
Input common-mode voltage	Over recommended operating conditions	1.43	1.5	1.57	V
DNR	–60-dB full-scale input applied at line inputs, A-weighted	90	93		dBA
THD + N	1-kHz, –4-dB full-scale input	–75	–80		dB
PSRR	1 kHz, 100 mVpp on AVDD	51	57		dB
Channel separation	1 kHz	–80	–90		dB
Input resistance		14.6	18.33	22	kΩ
Input capacitance			10		pF
DAC Channel/DAC Output	1-kHz sine-wave input, Load = 10 kΩ, 10 pF				
Full-scale output voltage (0 dB)		0.81	0.9		Vrms
Gain error		–10		10	%
Output common mode	Over recommended operating conditions	1.43	1.5	1.57	V
DNR	–60-dB full-scale input applied at line inputs, A-weighted	95	97		dBA
THD + N	–1-dBFS input, 0-dB gain	–80	–90		dB
PSRR	1 kHz, 100 mVpp on AVDD, V_{GND} powered down	50	56		dB
Load capacitance					pF
Load resistance		10			kΩ
Channel separation		–81	–84		dB
DAC Channel/ Headphone Output	1-kHz sine-wave input, Load = 16 Ω, External series resistance = 16 Ω, Coupling capacitance = 47 µF				
Full-scale output voltage (0 dB)		0.72	0.9		Vrms
DNR	–60-dB full-scale input applied at Line inputs, A-weighted	80	90		dBA
THD + N	0-dBFS input, 0-dB gain	–50	–60		dB
PSRR	1 kHz, 100 mVpp on AVDD , V_{GND} powered down	48	54		dB
Maximum output power(2)			24		mW
Load capacitance				100	pF
Load resistance		16			Ω
Channel separation		–70	–80		dB

(1) When the TAS3208 is operated in slave mode, the internal analog clocks for ADC and DAC are derived from external MCLKIN input. In this case, the analog performance will depend on MCLKIN quality (i.e., jitter, phase noise, etc.).
(2) 16-Ω series resistor required in L and R headphone outputs for short-circuit protection.

ELECTRICAL SPECIFICATIONS – ANALOG SECTIONS[1] (continued)

T_A = 25°C, AVDD = 3.3 V, DVDD = 3.3 V, Fs (audio) = 48 kHz, clock source from XTALI, AES17 filter, second-order 30-kHz low-pass filter (unless otherwise noted)

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
DAC Channel/Headphone Output	1-kHz sine-wave input, Load = 10 kΩ, 10 pF				
Full-scale output voltage (0 dB)		0.81	0.9		Vrms
DNR	–60-dB full-scale input applied at line inputs, A-weighted	80	90		dBA
THD + N	0-dBFS input, 0-dB gain	–70	–82		dB
PSRR	1 kHz, 100 mVpp on AVDD, V_{GND} powered down	48	54		dB
Channel separation		–70	–80		dB
Analog Mux in Bypass Mode	1-kHz sine-wave input, Load = 10 kΩ, 10 pF				
Mux switching noise	LINEIN inputs floating	–20		20	mV
Full-scale input voltage (0 dB)			1	1.15	Vrms
Input common-mode voltage		1.43	1.5	1.57	V
Load capacitance				20	pF
Load resistance		10			kΩ
Channel separation	Between Lch and Rch		–80		dB
	Between each line input		–80		dB
Full-scale output voltage (0 dB)		0.9	1	1.1	Vrms

ELECTRICAL CHARACTERISTICS

over recommended operating conditions (unless otherwise noted)

PARAMETER			TEST CONDITIONS	MIN	MAX	UNIT
V_{OH}	High-level output voltage	3.3-V TTL	I_{OH} = –4 mA	2.4		V
		1.8-V LVCMOS (XTL_OUT)	I_{OH} = –0.55 mA	1.44		
V_{OL}	Low-level output voltage	3.3-V TTL	I_{OL} = 4 mA		0.5	V
		1.8-V LVCMOS (XTL_OUT)	I_{OL} = 0.75 mA		0.396	
I_{OZ}	High-impedance output current	3.3-V TTL			±20	µA
I_{IL}	Low-level input current(1)	1.8-V LVCMOS (XTL_IN)	$V_I = V_{IL}$		±1	µA
		3.3-V TTL			±1	
I_{IH}	High-level input current(2)	1.8-V LVCMOS (XTL_IN)	$V_I = V_{IH}$		±1	µA
		3.3-V TTL			±1	
I_{DVDD}	Digital supply current		DSP clock = 135 MHz, LRCLKIN/LRCLKOUT = 48 KHz, XTALI = 24.288 MHz		200	mA
I_{AVDD}	Analog supply current		DSP clock = 135 MHz, LRCLKIN/LRCLKOUT = 48 KHz, XTALI = 24.288 MHz		28	mA
I_{DVDD}	Digital supply current		$\overline{RESET}$ = LOW		0.1	mA
I_{AVDD}	Analog supply current		$\overline{RESET}$ = LOW		5	mA

(1) Value given is for those input pins that connect to an internal pullup resistor, as well as an input buffer. For inputs that have a pulldown resistor or no resistor, I_{IL} = ±1 µA.
(2) Value given is for those input pins that connect to an internal pulldown resistor, as well as an input buffer. For inputs that have a pullup resistor or no resistor, I_{IH} = ± 1 µA.

TIMING REQUIREMENTS – MASTER CLOCK SIGNALS

over recommended operating conditions (see Figure 33)

			MIN	TYP	MAX	UNIT
f_{XTALI}	XTALI frequency (1/ t_{cyc1})(1)			24.576 (512 Fs)		MHz
t_{cyc1}	XTALI cycle time(2)			1/(512 Fs)		ns
f_{MCLKIN}	MCLKIN frequency (1/ t_{cyc2})			256 Fs		MHz
tw_{MCLKIN}	MCLKIN pulse duration(3)		0.4 × t_{cyc2}		0.6 × t_{cyc2}	ns
$f_{MCLKOUT}$	MCLKOUT frequency(1/ t_{cyc3})			256 Fs		MHz
$tr_{MCLKOUT}$	MCLKOUT rise time	C_L = 30 pF			10	ns
$tf_{MCLKOUT}$	MCLKOUT fall time	C_L = 30 pF			10	ns
$tw_{MCLKOUT}$	MCLKOUT pulse duration(4)		0.4 × t_{cyc3}		0.6 × t_{cyc3}	ns
	MCLKOUT jitter	XTALI master clock source		80		ps
$td_{MI–MO}$	Delay time, MCLKIN rising edge to MCLKOUT rising edge (5)	MCLKOUT = MCLKIN			17	ns

(1) Frequency tolerance is ±100 ppm (or better) at 25°C.
(2) t_{cyc1} = 1/ fX_{TALI}
(3) t_{cyc2} = 1/ f_{MCLKIN}
(4) t_{cyc3} = 1/ $f_{MCLKOUT}$
(5) When MCLKOUT is derived from MCLKIN, MCLKOUT jitter = MCLKIN jitter. MCLKOUT has the same duty cycle as MCLKIN when MCLKOUT = MCLKIN.

TIMING REQUIREMENTS – RESET

with respect to DVDD power good (see Figure 34)

		MIN	MAX	UNIT
$t_{pgw(L)}$	Minimum pulse duration, $\overline{RESET}$ low following DVDD = 3.3 V	100		ms

TIMING REQUIREMENTS – RESET

control signal parameters over recommended operating conditions (unless otherwise noted) (see Figure 35)

		MIN	TYP	MAX	UNIT
$tr_{DMSTATE}$	Time to outputs inactive		100		µs
tw_{RESET}	Pulse duration, $\overline{RESET}$ active	200			ns
$tr_{EMSTATE}$	Time to enable I^2C		<50		ms

TIMING REQUIREMENTS – SERIAL AUDIO PORT SLAVE MODE

over recommended operating conditions (unless otherwise noted) (see Figure 36)

			MIN	TYP	MAX	UNIT
f_{LRCLK}	Frequency, LRCLKIN (FS)		32		48	kHz
tw_{SCLKIN}	Pulse duration, SCLKIN high(1)		0.4 × t_{cyc}		0.6 × t_{cyc}	ns
f_{SCLKIN}	Frequency, SCLKIN			64 Fs		MHz
t_{cyc}	Cycle time, SCLKIN(1)				1/64 Fs	ns
t_{pd1}	Propagation delay, SCLKIN falling edge to SDOUT				16	ns
t_{su1}	Setup time, LRCLK to SCLKIN rising edge		10			ns
t_{h1}	Hold time, LRCLK from SCLKIN rising edge		5			ns
t_{su2}	Setup time, SDIN to SCLKIN rising edge		10			ns
t_{h2}	Hold time, SDIN from SCLKIN rising edge		5			ns
t_{pd2}	Propagation delay, SCLKIN falling edge to SCLKOUT falling edge	SCLKOUT = SCLKIN			15	ns

(1) $t_{cyc} = 1/ f_{SCLKIN}$

TIMING REQUIREMENTS – SERIAL AUDIO PORT MASTER MODE

over recommended operating conditions (unless otherwise noted) (see Figure 37)

			MIN	TYP	MAX	UNIT
f_{LRCLK}	Frequency, LRCLKOUT			48		kHz
tr_{LRCLK}	Rise time, LRCLKOUT	C_L = 30 pF			12	ns
tf_{LRCLK}	Fall time, LRCLKOUT	C_L = 30 pF			12	ns
$f_{SCLKOUT}$	Frequency, SCLKOUT(1)			64 Fs		MHz
$tr_{SCLKOUT}$	Rise time, SCLKOUT	C_L = 30 pF			12	ns
$tf_{SCLKOUT}$	Fall time, SCLKOUT	C_L = 30 pF			12	ns
t_{pd1}	Propagation delay, SCLKOUT falling edge to LRCLKOUT edge				5	ns
t_{pd2}	Propagation delay, SCLKOUT falling edge to SDOUT1–2				5	ns
t_{su}	Setup time, SDIN to SCLKOUT rising edge		25			ns
t_h	Hold time, SDIN from SCLKOUT rising edge		30			ns

(1) Typical duty cycle is 50/50.

TIMING CHARACTERISTICS – SPDIF INTERFACE

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
Fs	Encoded data sampling rate		32		48	kHz
R_{spdif}	SPDIF signal bit rate			128 Fs		MHz
UI	Unit interval			$1/R_{spdif}$		ns
T_{LO}/T_{HI}	Low/high periods		1 UI		3 UI	ns
V_{OH}	High-level output voltage	3.3-V TTL, I_{OH} = –4 mA	2.4			V
V_{OL}	Low-level output voltage	3.3-V TTL, I_{OL} = 4 mA			0.5	V

I^2C INTERFACE AND I/O CHARACTERISTICS OF SDA AND SCL BUS LINES FOR STANDARD-MODE AND FAST-MODE I^2C BUS DEVICES

See Figure 38

		STANDARD MODE		FAST MODE		UNIT
		MIN	MAX	MIN	MAX	
f_{SCL}	SCL clock frequency	0	100	0	400 (1)	kHz
$t_{HD;STA}$	Hold time, (repeated) Start condition. After this period, the first clock pulse is generated.	4		0.6		µs
t_{LOW}	LOW period of SCL clock	4.7		1.3		µs
tHI_{GH}	HIGH period of SCL clock	4		0.6		µs
$t_{su;STA}$	Setup time, repeated Start condition	4.7		0.6		µs
$t_{su;DAT}$	Data setup time	250		100 (2)		ns
t_r	Rise time, both SDA and SCL signals		1000	20 + 0.1 × C_b (3)	300	ns
t_f	Fall time, both SDA and SCL signals		300	20 + 0.1 × C_b (3)	300	ns
$t_{su;STO}$	Setup time, Stop condition	4		0.6		µs
t_{BUF}	Bus free time between Stop and Start condition	4.7		1.3		µs
C_b	Capacitive load for each bus line		400		400	pF
V_{nL}	Noise margin at LOW level for each connected device (including hysteresis)	0.1 × V_{DD}		0.1 × V_{DD}		V
V_{nH}	Noise margin at HIGH level for each connected device (including hysteresis)	0.2 × V_{DD}		0.2 × V_{DD}		V
V_{hys}	Hysteresis of Schmitt-trigger inputs			0.05 × V_{DD}		V
t_{SP}	Pulse width of spikes that must be suppressed by the input filter			0	50	ns
I_i	Input current each I/O pin with an input voltage between 0.1 × V_{DD} and 0.9 × V_{DD} max	–10	10	–10 (4)	10 (4)	µA
C_i	Capacitance for each I/O pin		10		10	pF
t_{of}	Output fall time from V_{IH}min to V_{IL}max, with a bus capacitance from 10 pF to 400 pF		250 (5)	7 + 0.1 × C_b (3)	250 (5)	ns

(1) In Master mode, the maximum I^2C clock rate is 375 kHz.
(2) A Fast-mode I^2C bus device can be used in a Standard-mode I^2C bus system, but the requirement $t_{SU;DAT} \geq 250$ ns must then be met. This will automatically be the case if the device does not stretch the LOW period of the SCL signal. If such a device does stretch the LOW period of the SCL signal, it must output the next data bit to the SDA line.
(3) C_b = Total capacitance of one bus line in pF
(4) I/O pins of Fast-mode devices must not obstruct the SDA and SCL lines if V_{DD} is switched off.
(5) The maximum t_f for the SDA and SCL bus lines (300 ns) is longer than the specified maximum tof for the output stages (250 ns). This allows series protection resistors (Rs) to be connected between the SDA/SCL pins and the SDA/SCL bus lines, without exceeding the maximum specified t_f.

PARAMETER MEASUREMENT INFORMATION

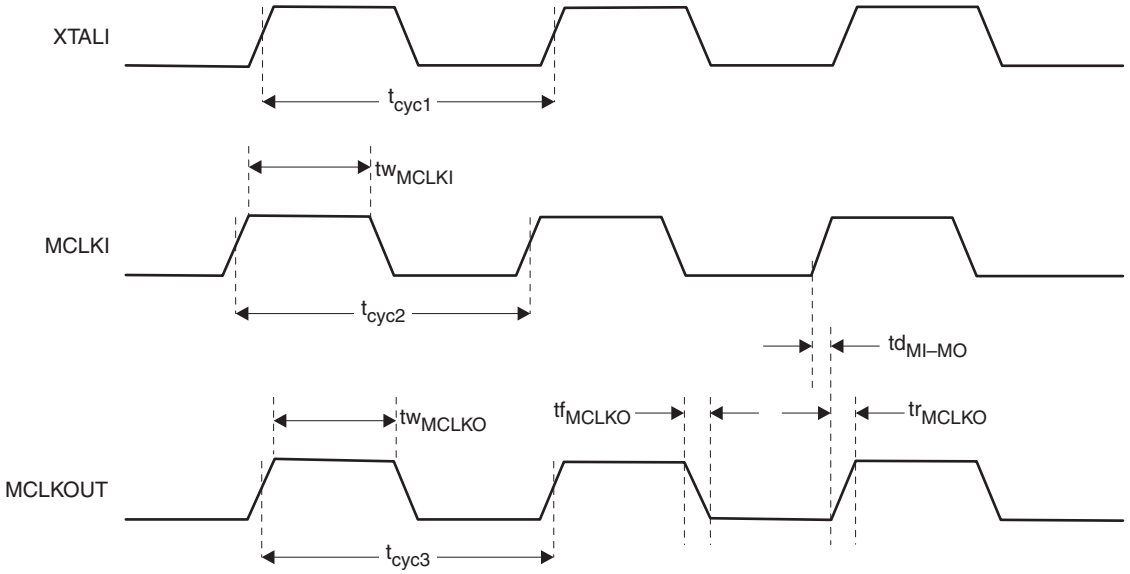


Figure 33. Master Clock Signals Timing

Figure 34. Reset Timing During Power On

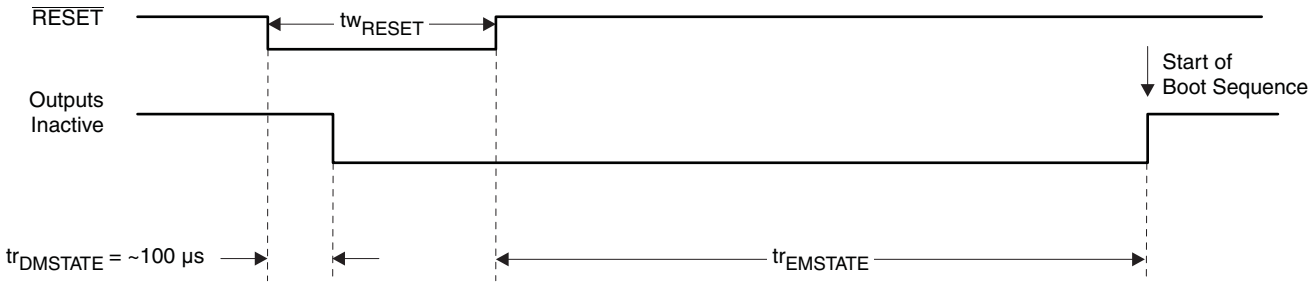


Figure 35. Reset Timing

PARAMETER MEASUREMENT INFORMATION (continued)

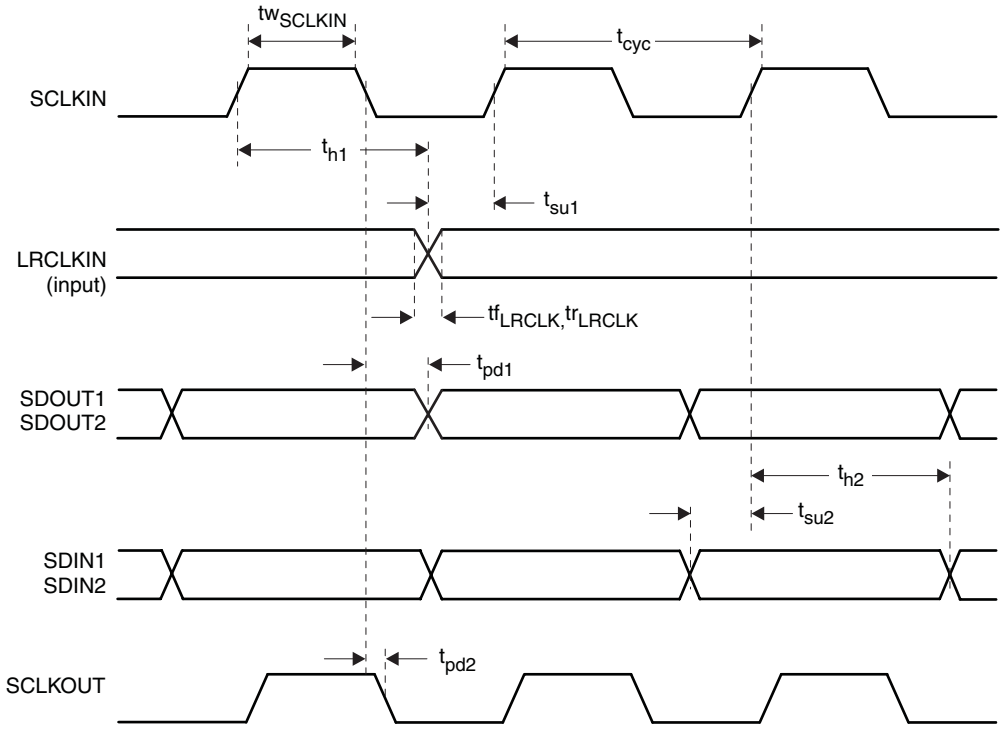


Figure 36. Serial Audio Port Slave-Mode Timing

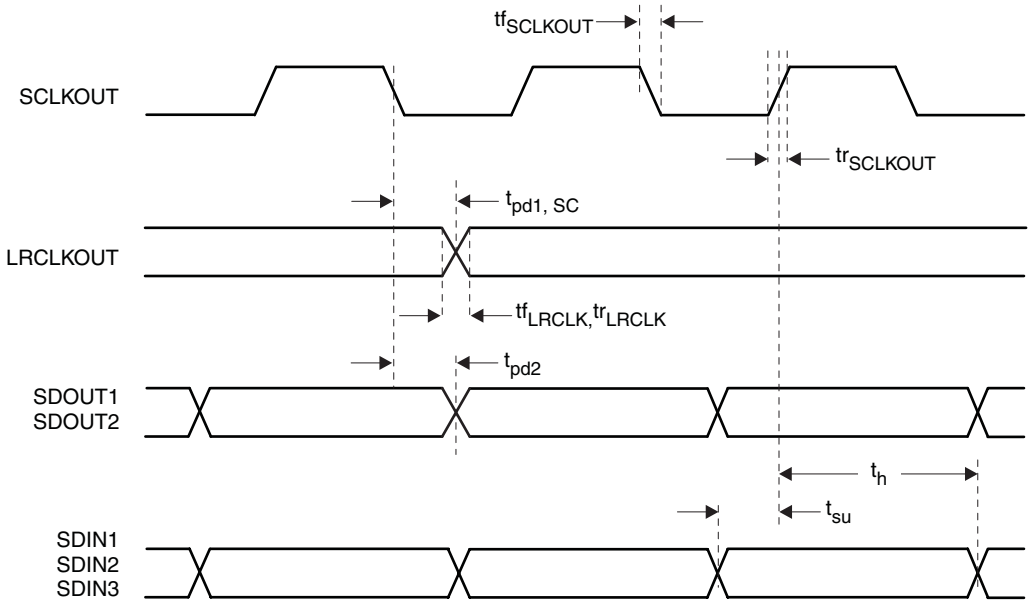


Figure 37. Serial Audio Port Master-Mode Timing

PARAMETER MEASUREMENT INFORMATION (continued)

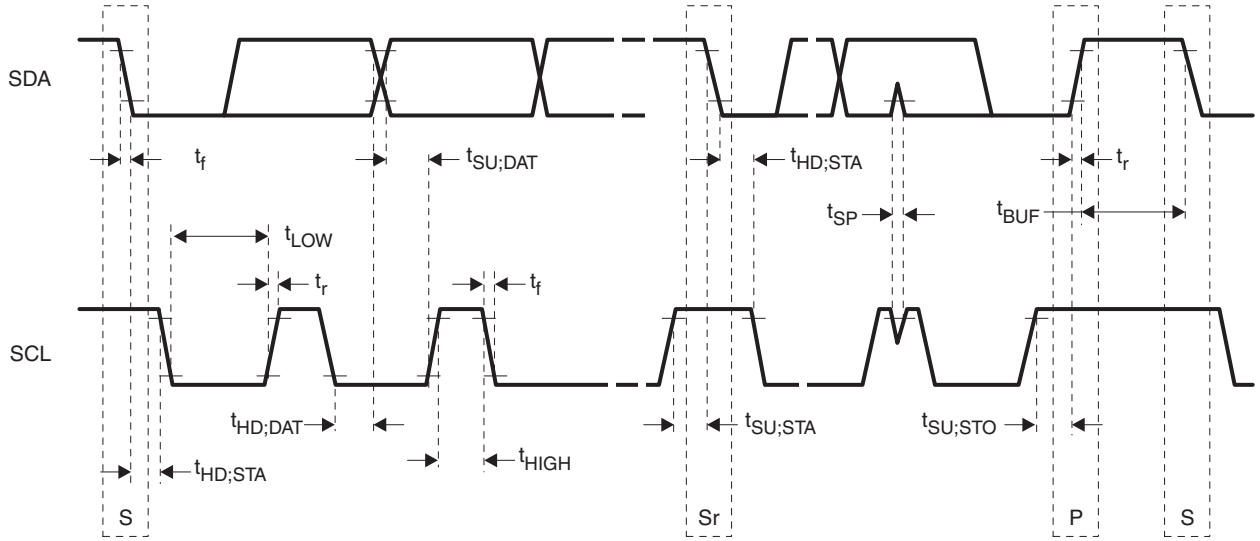


Figure 38. I²C SCL and SDA Timing

Master I²C Load RAM Block Formats

This section describes the format of the data that is stored in an external memory device and downloaded to the TAS3208 via the master I²C bus.

Master I²C Memory Block Header

Table 11. 1 Memory Block Header

STARTING BYTE	DATA BLOCK FORMAT	SIZE	NOTES
0	Checksum MSB	2 bytes	Checksum of byte 2 through N + 12
	Checksum LSB		
2	Header ID byte 1 = 0x00	2 bytes	Must be 0x001F
	Header ID byte 2 = 0x1F		
4	Memory to be loaded	1 byte	0x00: Microprogram RAM or termination header 0x01: Micro external data RAM 0x02: DSP program RAM 0x03 : DSP coefficient RAM 0x04: DSP data RAM 0x05–0x0F: Reserved
5	0x00	1 byte	Unused
6	Start memory address MSB	2 bytes	If this is a termination header, this value is 0000.
	Start memory address LSB		
8	Total number of byte transferred MSB	2 bytes	Header size (12) + data byte + last checksum byte. If this is a termination header, this value is 0000.
	Total number of byte transferred LSB		
10	0x00	1 byte	Unused
11	0x00	1 byte	Unused

Master I²C Download Memory Block Structure

Table 12. 1 M8051 MCU Program RAM and External Data RAM Block Structure

STARTING BYTE	DATA BLOCK FORMAT	SIZE	VALUE	NOTES
0	Checksum MSB	2 bytes		Checksum of byte 2 through N + 12
	Checksum LSB			
2	Header ID byte 1	2 bytes	0x00	Must be 0x001F
	Header ID byte 2		0x1F	
4	Memory to be loaded	1 byte	0x00 or 0x01	Microprogram RAM or micro external data RAM
5	0x00	1 byte	0x00	Unused
6	Start memory address MSB	2 bytes		If this is a termination header, this value is 0000
	Start memory address LSB			
8	Total number of byte transferred MSB	2 bytes		Header (12) + data (N) + checksum (4)
	Total number of byte transferred LSB			
10	0x00	1 byte	0x00	Unused
11	0x00	1 byte	0x00	Unused
12	Data byte 1 (LSB)	4 bytes		1–4 microprocessor byte
	Data byte 2			
	Data byte 3			
	Data byte 4 (MSB)			
16	Data byte 5 (LSB)	4 bytes		5–8 microprocessor byte
	Data byte 6			
	Data byte 7			
	Data byte 8 (MSB)			
⋮	⋮	⋮	⋮	⋮
N + 12	0x00	4 bytes		Repeated checksum byte 2 through N +11
	0x00			
	Checksum MSB			
	Checksum LSB			

Table 13. DSP Program RAM Block Structure

STARTING BYTE	DATA BLOCK FORMAT	SIZE	VALUE	NOTES
0	Checksum MSB	2 bytes		Checksum of byte 2 through N + 12
	Checksum LSB			
2	Header ID byte 1	2 bytes	0x00	Must be 0x001F
	Header ID byte 2		0x1F	
4	Memory to be loaded	1 byte	0x02	Microprogram RAM or micro external data RAM
5	0x00	1 byte	0x00	Unused
6	Start memory address MSB	2 bytes		If this is a termination header, this value is 0000.
	Start memory address LSB			
8	Total number of byte transferred MSB	2 bytes		Header (12) + data (N) + checksum (4)
	Total number of byte transferred LSB			
10	0x00	1 byte	0x00	Unused
11	0x00	1 byte	0x00	Unused
12	Program byte 1 (LSB)	7 bytes		Program word 1 D7–D0
	Program byte 2			D15–D8
	Program byte 3			D23–D16
	Program byte 4			D31–D24
	Program byte 5			D39–D32
	Program byte 6			D47–D40
	Program byte 7 (MSB)			D55–D48
19	Program byte 8 (LSB)	7 bytes		Program word 2
	Program byte 9			
	Program byte 10			
	Program byte 11			
	Program byte 12			
	Program byte 13			
	Program byte 14 (MSB)			
⋮	⋮	⋮	⋮	⋮
N + 12	0x00	7 bytes		Repeated checksum byte 2 through N +11
	0x00			
	0x00			
	0x00			
	0x00			
	Checksum MSB			
	Checksum LSB			

Table 14. DSP Coefficient RAM Block Structure

STARTING BYTE	DATA BLOCK FORMAT	SIZE	VALUE	NOTES
0	Checksum MSB	2 bytes		Checksum of byte 2 through N + 12
	Checksum LSB			
2	Header ID byte 1	2 bytes	0x00	Must be 0x001F
	Header ID byte 2		0x1F	
4	Memory to be loaded	1 byte	0x03	Microprogram RAM or micro external data RAM
5	0x00	1 byte	0x00	Unused
6	Start memory address MSB	2 bytes		If this is a termination header, this value is 0000.
	Start memory address LSB			
8	Total number of byte transferred MSB	2 bytes		Header (12) + data (N) + checksum (4)
	Total number of byte transferred LSB			
10	0x00	1 byte	0x00	Unused
11	0x00	1 byte	0x00	Unused
12	Data byte 1 (LSB)	4 bytes		Coefficient word 1 D7–D0
	Data byte 2			D15–D8
	Data byte 3			D23–D16
	Data byte 4 (MSB)			D31–D24
16	Data byte 5 (LSB)	4 bytes		Coefficient word 2
	Data byte 6			
	Data byte 7			
	Data byte 8 (MSB)			
⋮	⋮	⋮	⋮	⋮
N + 12	0x00	4 bytes		Repeated checksum byte 2 through N +11
	0x00			
	Checksum MSB			
	Checksum LSB			

Table 15. DSP Data RAM Block Structure

STARTING BYTE	DATA BLOCK FORMAT	SIZE	VALUE	NOTES
0	Checksum MSB	2 bytes		Checksum of byte 2 through N + 12
	Checksum LSB			
2	Header ID byte 1	2 bytes	0x00	Must be 0x001F
	Header ID byte 2		0x1F	
4	Memory to be loaded	1 byte	0x04	Microprogram RAM or micro external data RAM
5	0x00	1 byte	0x00	Unused
6	Start memory address MSB	2 bytes		If this is a termination header, this value is 0000.
	Start memory address LSB			
8	Total number of byte transferred MSB	2 bytes		Header (12) + data (N) + checksum (4)
	Total number of byte transferred LSB			
10	0x00	1 byte	0x00	Unused
11	0x00	1 byte	0x00	Unused
12	Data byte 1 (LSB)	6 bytes		Data word 1 D7–D0
	Data byte 2			D15–D8
	Data byte 3			D23–D16
	Data byte 4 (MSB)			D31–D24
	Data byte 5			D39–D32
	Data byte 6 (MSB)			D47–D40
18	Data byte 7 (LSB)	6 bytes		Data word 2
	Data byte 8			
	Data byte 9			
	Data byte 10			
	Data byte 11			
	Data byte 12 (MSB)			
⋮	⋮	⋮	⋮	⋮
N + 12	0x00	6 bytes		Repeated checksum byte 2 through N +11
	0x00			
	Checksum MSB			
	Checksum LSB			

Slave I²C Load RAM Block Formats

The slave I²C bus permits the system controller to load the TAS3208 memories as an alternative to using the master download from an external memory device via the I²C master bus. The transfer is performed by writing to two I²C registers (0x04 and 0x05). The first register holds the header information, and the second register holds eight bytes of data. Figure 39 shows the I²C slave download flow.

I²C slave download register format are described in Table 16 to Table 20. The I²C slave download process is terminated when a termination header with zero-length byte count field is received.

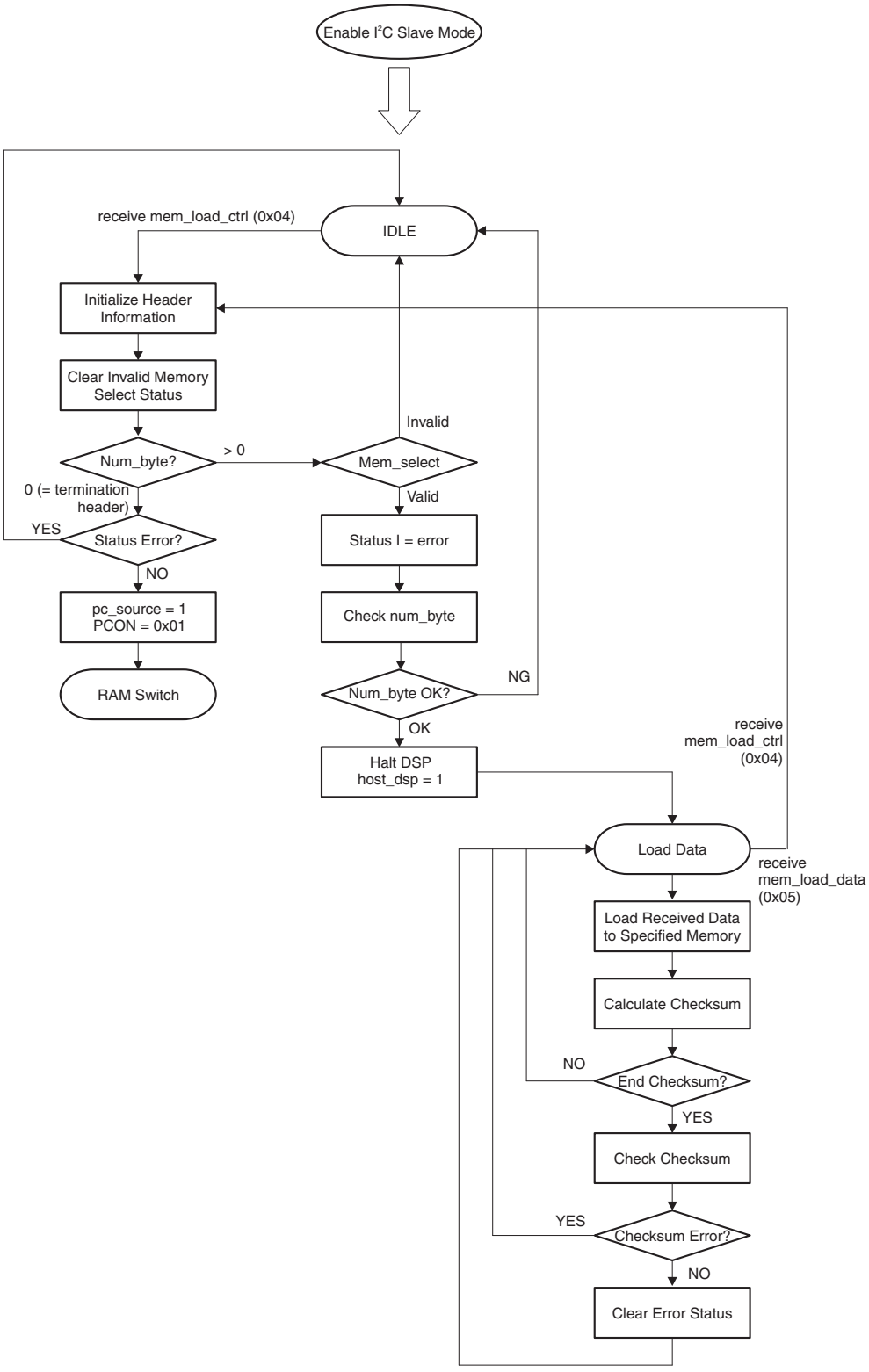


Figure 39. I²C Slave Download Flow

Table 16. M8051 Microcontroller Program RAM and External Data RAM Block Structure(1)

REGISTER	BYTE	DATA BLOCK FORMAT	CALC CHECKSUM	TOTAL NO. BYTES	NOTE
Control 0x04	1	Checksum MSB			
	2	Checksum LSB			
	3	Memory to be loaded 0x00 or 0x01			
	4	0x00			
	5	Start memory address MSB			
	6	Start memory address LSB			
	7	Total number of byte transferred MSB			
	8	Total number of byte transferred LSB			
Data 0x05	1	Datum 1 D7–D0			
	2	Datum 2 D7–D0			
	3	Datum 3 D7–D0			
	4	Datum 4D7–D0			
	5	Datum 5 D7–D0			
	6	Datum 6 D7–D0			
	7	Datum 7 D7–D0			
	8	Datum 8 D7–D0			
Data 0x05	1	Datum 9 D7–D0			
	2	Datum 10 D7–D0			
	3	Datum 11D7–D0			
	4	Datum 12 D7–D0			
	5	Datum 13 D7–D0			
	6	Datum 14 D7–D0			
	7	Datum 15 D7–D0			
	8	Datum 16 D7–D0			
Data 0x05	1	Datum N-3 D7–D0			
	2	Datum N-2 D7–D0			If the last data register datum is less than 6 byte, zero data should be filled.
	3	Datum N-1 D7–D0			
	4	Datum N D7–D0			
	5	0x00			Should be zero
	6	0x00			
	7	Checksum MSB			End checksum is always located here.
	8	Checksum LSB			

(1) Shades cells indicate the values included in the checksum/total number of bytes calculation.

Table 17. DSP Program RAM Block Structure(1)

REGISTER	BYTE	DATA BLOCK FORMAT	CALC CHECKSUM	TOTAL NO. BYTES	NOTE
Control 0x04	1	Checksum MSB			
	2	Checksum LSB			
	3	Memory to be loaded 0x02			
	4	0x00			
	5	Start memory address MSB			
	6	Start memory address LSB			
	7	Total number of byte transferred MSB			
	8	Total number of byte transferred LSB			
Data 0x05	1	0x00			
	2	D55–D48			
	3	D47–D40			
	4	D39–D32			
	5	D31–D24			Program word 1
	6	D23–D16			
	7	D15–D8			
	8	D7–D0			
Data 0x05	1	0x00			
	2	D55–D48			
	3	D47–D40			
	4	D39–D32			
	5	D31–D24			Program word 2
	6	D23–D16			
	7	D15–D8			
	8	D7–D0			
Data 0x05	1	0x00			
	2	0x00			
	3	0x00			
	4	0x00			Should be zero
	5	0x00			
	6	0x00			
	7	Checksum MSB			End checksum is always located here.
	8	Checksum LSB			

(1) Shades cells indicate the values included in the checksum/total number of bytes calculation.

Table 18. DSP Coefficient RAM Block Structure(1)

REGISTER	BYTE	DATA BLOCK FORMAT	CALC CHECKSUM	TOTAL NO. BYTES	NOTE
Control 0x04	1	Checksum MSB			
	2	Checksum LSB			
	3	Memory to be loaded 0x03			
	4	0x00			
	5	Start memory address MSB			
	6	Start memory address LSB			
	7	Total number of byte transferred MSB			
	8	Total number of byte transferred LSB			
Data 0x05	1	D31–D24			Coefficient word 1
	2	D23–D16			
	3	D15–D8			
	4	D7–D0			
	5	D31–D24			Coefficient word 2
	6	D23–D16			
	7	D15–D8			
	8	D7–D0			
Data 0x05	1	D31–D24			Coefficient word 3
	2	D23–D16			
	3	D15–D8			
	4	D7–D0			
	5	D31–D24			Coefficient word 4
	6	D23–D16			
	7	D15–D8			
	8	D7–D0			
Data 0x05	1	D31–D24			Coefficient word N or zero
	2	D23–D16			
	3	D15–D8			
	4	D7–D0			
	5	0x00			Should be zero
	6	0x00			
	7	Checksum MSB			End checksum is always located here.
	8	Checksum LSB			

(1) Shades cells indicate the values included in the checksum/total number of bytes calculation.

Table 19. DSP Data Block Structure(1)

REGISTER	BYTE	DATA BLOCK FORMAT	CALC CHECKSUM	TOTAL NO. BYTES	NOTE
Control 0x04	1	Checksum MSB			
	2	Checksum LSB			
	3	Memory to be loaded 0x04			
	4	0x00			
	5	Start memory address MSB			
	6	Start memory address LSB			
	7	Total number of byte transferred MSB			
	8	Total number of byte transferred LSB			
Data 0x05	1	0x00			
	2	0x00			
	3	D47–D40			Coefficient word 1
	4	D39–D32			
	5	D31–D24			
	6	D23–D16			Coefficient word 2
	7	D15–D8			
	8	D7–D0			
Data 0x05	1	0x00			
	2	0x00			
	3	D47–D40			Coefficient word 3
	4	D39–D32			
	5	D31–D24			
	6	D23–D16			Coefficient word 4
	7	D15–D8			
	8	D7–D0			
Data 0x05	1	0x00			
	2	0x00			
	3	0x00			
	4	0x00			Should be zero
	5	0x00			
	6	0x00			
	7	Checksum MSB			End checksum is always located here.
	8	Checksum LSB			

(1) Shades cells indicate the values included in the checksum/total number of bytes calculation.

Table 20. Termination Header Block Structure(1)

REGISTER	BYTE	DATA BLOCK FORMAT	CALC CHECKSUM	TOTAL NO. BYTES	NOTE
Control 0x04	1	Checksum MSB			00
	2	Checksum LSB			00
	3	Memory to be loaded			00
	4	0x00			00
	5	Start memory address MSB			00
	6	Start memory address LSB			00
	7	Total number of byte transferred MSB			00
	8	Total number of byte transferred LSB			00

(1) Shades cells indicate the values included in the checksum/total number of bytes calculation.

I²C Register Map

The I²C register map for ROM advanced code is described in Table 21.

Table 21. I²C Register Map(1)

SUB ADDRESS	REGISTER	BYTES	CONTENTS	DEFAULT VALUE
0x00	SAP/Clock Setting	4	See SAP/Clock Setting Register	
0x01	I²C M and N	4	u(31:24), u(23:16), u(15:8), u(7)M(6:3)N(2:0)	0x00, 0x00, 0x00, 0x00
0x02	Status	8	See Status Register	0x00, 0x00, 0x00, 0x00 0x00, 0x00, 0x00, 0x00
0x03	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x04	I²C RAM Load Control	8	See I2C RAM Load Control Register	0x00, 0x00, 0x00, 0x00 0x00, 0x00, 0x00, 0x00
0x05	I²C RAM Load Data	8	See I2C RAM Load Data Register	0x00, 0x00, 0x00, 0x00 0x00, 0x00, 0x00, 0x00
0x06	PEEK/POKE Control	4	See PEEK/POKE Control Register	0x00, 0x00, 0x00, 0x00
0x07	PEEK/POKE Data	8	See PEEK/POKE Data Register	0x00, 0x00, 0x00, 0x00 0x00, 0x00, 0x00, 0x00
0x08	Silicon Version	4	ver(31:24), ver(23:16), ver(15:8), ver(7:0)	0x00, 0x00, 0x00, 0x02
0x09	Mute Control	4	See Mute Control Register	0x00, 0x00, 0x00, 0x00
0x0a	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x0b	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x0c	GPIO Control	4	See GPIO Control Register	0x00, 0x00, 0x00, 0x00
0x0d	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x0e	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x0f	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x10	Powerdown Control	4	See Powerdown Control Register	0x00, 0x00, 0x00, 0x00
0x11	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x12	A-MUX Control	4	See A-MUX Control Register	0x00, 0x00, 0x00, 0x00
0x13	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x14	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x15	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00

(1) Shades cells indicate common to basic and advanced modes. Unshaded cells indicate advanced mode only.

Table 21. I²C Register Map(1) (continued)

SUB ADDRESS	REGISTER	BYTES	CONTENTS	DEFAULT VALUE
0x16	SPDIF Control	4	See SPDIF Control Register	0x00, 0x00, 0x00, 0x00
0x17	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x18	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x19	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x1a	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x1b	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x1c	Reserved	8	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x01 0x47, 0xae, 0x00, 0x00
0x1d	DC Dither	4	See DC Dither Register	0x00, 0x00, 0x00, 0x01
0x1e	DSP Program Start Address	4	See DSP Program Start Address Register	0x00, 0x00, 0x00, 0x00
0x1f	Reserved	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x20	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x21	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x22	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x23	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x24	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x25	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x26	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x27	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x28	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x29	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x2a	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x2b	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x2c	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x2d	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x2e	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x2f	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x30	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x31	Unused	4	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x32	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x33	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x34	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x35	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x36	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x37	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x38	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x39	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x3a	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x3b	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x3c	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0x3d	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0xfe	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00
0xff	Unused	16	u(31:24), u(23:16), u(15:8), u(7:0)	0x00, 0x00, 0x00, 0x00

SAP/Clock Setting Register (0x00)

The SAP/Clock Setting register is used to configure the device as a clock master/slave, as well as specify the desired format of the digital audio ports. This register is four bytes in length.

Table 22. SAP/Clock Setting Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	0	0	0	0	0	0	0		Unused
								CM/S	Clock master/slave select
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
									Unused
								ON	SAP output normalization
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0								Unused
		OW1	OW0						Digital audio output word size
				0	0				Unused
						IW1	IW0		Digital audio input word size
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	0								Unused
		OM1	OM0						Digital audio output format
				0	0				Unused
						IM1	IM0		Digital audio input format

Table 23. Clock Master/Slave Select(1)

CLOCK MASTER/SLAVE SELECT	CMS
Master	*1*
Slave	0

(1) Default values are shown in italics.

Table 24. Digital Audio Port Normalization(1)

DIGITAL AUDIO PORT NORMALIZATION	ON
Enable	*1*
Disable	0

(1) Default values are shown in italics.

Bits 9–8 (IW1 and IW0) define the data word size for the input SAP. Bits 13–12 (OW1 and OW0) define the data word size for the output SAP.

Table 25. Audio Data Word Size(1)

DIGITAL AUDIO I/O WORD SIZE	IW1/OW1	IW0/OW0
16 bit	0	0
20 bit	0	1
24 bit	*1*	*0*
–	1	1

(1) Default values are shown in italics.

Table 26. Audio Data Format(1)

DIGITAL AUDIO I/O FORMAT	IM1/OM1	IM0/OM0
Left-justified	0	0
Right-justified	0	1
I²S	*1*	*0*
–	1	1

(1) Default values are shown in italics.

Status Register (0x02)

The Status register provide memory load information. When a memory load error for a particular memory occurs, the memory load error bit for that memory is set to 1. When a memory load is successful for a particular memory, the memory load error bit for that memory is set to 0. The host must check this load status after memory load. The host can clear all load error status by writing 0 to bits D40–D32 of this register.

Table 27. Status Register

BIT	63	62	61	60	59	58	57	56	DESCRIPTION
	0	0	0	0	0	0	0	0	Reserved
BIT	**55**	**54**	**53**	**52**	**51**	**50**	**49**	**48**	
	0	0	0	0	0	0	0		Reserved
BIT	**47**	**46**	**45**	**44**	**43**	**42**	**41**	**40**	
	0	0	0	0	0	0	0	0	Unsused
BIT	**39**	**38**	**37**	**36**	**35**	**34**	**33**	**32**	
	x	x	x	x	x	x	x	1	M8051 program memory load error
	x	x	x	x	x	x	1	x	M8051 external memory load error
	x	x	x	x	x	1	x	x	DSP program memory load error
	x	x	x	x	1	x	x	x	DSP coefficient memory load error
	x	x	x	1	x	x	x	x	DSP data memory load error
	x	1	x	x	x	x	x	x	Invalid memory select
	1	x	x	x	x	x	x	x	End of load header error
	1	1	1	1	1	1	1	1	No EEPROM
	0	0	0	0	0	0	0	0	No error
BIT	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	
	0	0	0	0	0	0	0		Reserved
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0		Reserved
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0								Reserved
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	0								Reserved
		ABSY							Analog busy flag
			0						Reserved
				0					Reserved
					0				Reserved
						0			Reserved
							BUSE		I²C bus error
								0	Reserved

Bits 40–32 define the memory load error status on EEPROM download and slave download.

Table 28. Analog Busy(1)

ANALOG BUSY FLAG	ABSY
Analog is busy	1
Analog not busy	*0*

(1) Default values are shown in italics.

Analog control sequence takes time (maximum approximately 500 ms for headphone power up). This busy flag indicates whether the analog control sequence is running or not.

Table 29. I²C Bus Error(1)

I²C BUS ERROR	BUSE
Bus error	1
No bus error	*0*

(1) Default values are shown in italics.

If an I²C bus error occurs, this flag will be set. Only the host microcontroller can clear this flag by writing 0 to this bit. I²C bus error status is read from ESFR 0xC5, bit 6, and is cleared by ESFR 0xC7, bit 6.

I²C RAM Load Control and Data Registers (0x04 and 0x05)

The I²C memory load port permits the system controller to load the TAS3208 memories as an alternative to having the TAS3208 load its memory from an external EEPROM.

The transfer is performed by writing to two I²C registers. The first register is a 8-byte register than holds the checksum, memory to be written, starting address, and number of data bytes to be transferred. The second register holds eight bytes of data.

The memory load operation starts with the first register being set. Then the data is written into the second register using the format shown. After the last data byte is written into the second register, an additional two bytes are written, which constrain the 2-byte checksum. At that point, the transfer is complete and status of the operation is reported in the Status register.

NOTE

Once the microprogram memory has been loaded, further updates to this memory are inhibited until the device is reset.

When the first I²C slave download register is written by the system controller, the TAS3208 updates the Status register by setting a error bit to indicate an error for the memory type that is being loaded. This error bit is reset when the operation complete and a valid checksum has been received.

For example, when the microprogram memory is being loaded, the TAS3208 will set a microprogram memory error indication in the Status register at the start of the sequence. When the last byte of the microprogram memory and checksum is received, the TAS3208 will clear the microprogram memory error indication. This enables the TAS3208 to preserve any error status indications that occur as a result of incomplete transfers of data/ checksum error during a series of data and program memory load operations.

The checksum is always contained in the last two bytes of the data block.

The I²C slave download is terminated when a termination header with a zero-length byte count field is received.

Table 30. I²C RAM Load Control Register (0x04)

BYTE	DATA BLOCK FORMAT	SIZE	NOTES
1–2	Checksum code	2 bytes	Checksum of bytes 2 through N + 8, If this is a termination header, this value is 00 00.
3	Memory to be loaded	1 byte	0: Microprogram memory 1: Micro external data memory 2: DSP program memory 3: DSP coefficient memory 4: DSP data memory 5–15: Reserved
4	Unused	1 byte	Reserved
6–7	Starting TAS3208 memory address	2 bytes	If this is a termination header, this value is 00 00.
7–8	Number of data bytes to be transferred	2 bytes	If this is a termination header, this value is 00 00.

Table 31. I²C RAM Load Data Register (0x05)

BYTE	8-BIT DATA	24-BIT DATA	28-BIT DATA	48-BIT DATA	55-BIT DATA
1	Datum 1 D7–D0		XXXX D27–D24		
2	Datum 2 D7–D0	D23–D16	D23–D16		X D54–D48
3	Datum 3 D7–D0	D15–D8	D15–D8	D47–D40	D47–D40
4	Datum 4 D7–D0	D7–D0	D7–D0	D39–D32	D39–D32
5	Datum 5 D7–D0		XXXX D27–D24	D31–D24	D31–D24
6	Datum 6 D7–D0	D23–D16	D23–D16	D23–D16	D23–D16
7	Datum 7 D7–D0	D15–D8	D15–D8	D15–D8	D15–D8
8	Datum 8 D7–D0	D7–D0	D7–D0	D7–D0	D7–D0

PEEK/POKE Control and Data Registers (0x06 and 0x07)

The PEEK/POKE Control (Table 32) and PEEK/POKE Data (Table 33) registers allow the user to access the internal resources of TAS3208. Figure 40 shows the I²C transaction for the PEEK/POKE registers.

Table 32. PEEK/POKE Control Register (0x06)

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	0	0	0	0	0	0	0	0	Unused
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0	1	DSP coefficient memory load error
	0	0	0	0	0	0	1	0	DSP data memory load error
	0	0	0	0	0	0	1	1	DSP delay memory
	0	0	0	0	0	1	0	0	M8051 internal data memory
	0	0	0	0	0	1	0	1	M8051 external data memory
	0	0	0	0	0	1	1	0	Extended special function registers
	0	0	0	0	0	1	1	1	M8051 program memory
	0	0	0	0	1	0	0	0	DSP program memory
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0	0	0	0	0	0	0	0	Memory address MSB
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	0	0	0	0	0	0	0	0	Memory address LSB

Table 33. PEEK/POKE Data Register (0x07)

BIT	63	62	61	60	59	58	57	56	DESCRIPTION
	D63	D62	D61	D60	D59	D58	D57	D56	Data to be read or written
BIT	**55**	**54**	**53**	**52**	**51**	**50**	**49**	**48**	
	D55	D54	D53	D52	D51	D50	D49	D48	Data to be read or written
BIT	**47**	**46**	**45**	**44**	**43**	**42**	**41**	**40**	
	D47	D46	D45	D44	D43	D42	D41	D40	Data to be read or written
BIT	**39**	**38**	**37**	**36**	**35**	**34**	**33**	**32**	
	D39	D38	D37	D36	D35	D34	D33	D32	Data to be read or written
BIT	**31**	**30**	**29**	**28**	**27**	**26**	**25**	**24**	
	D31	D30	D29	D28	D27	D26	D25	D24	Data to be read or written
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	D23	D22	D21	D20	D19	D18	D17	D16	Data to be read or written
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	D15	D14	D13	D12	D11	D10	D9	D8	Data to be read or written
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	D7	D6	D5	D4	D3	D2	D1	D0	Data to be read or written

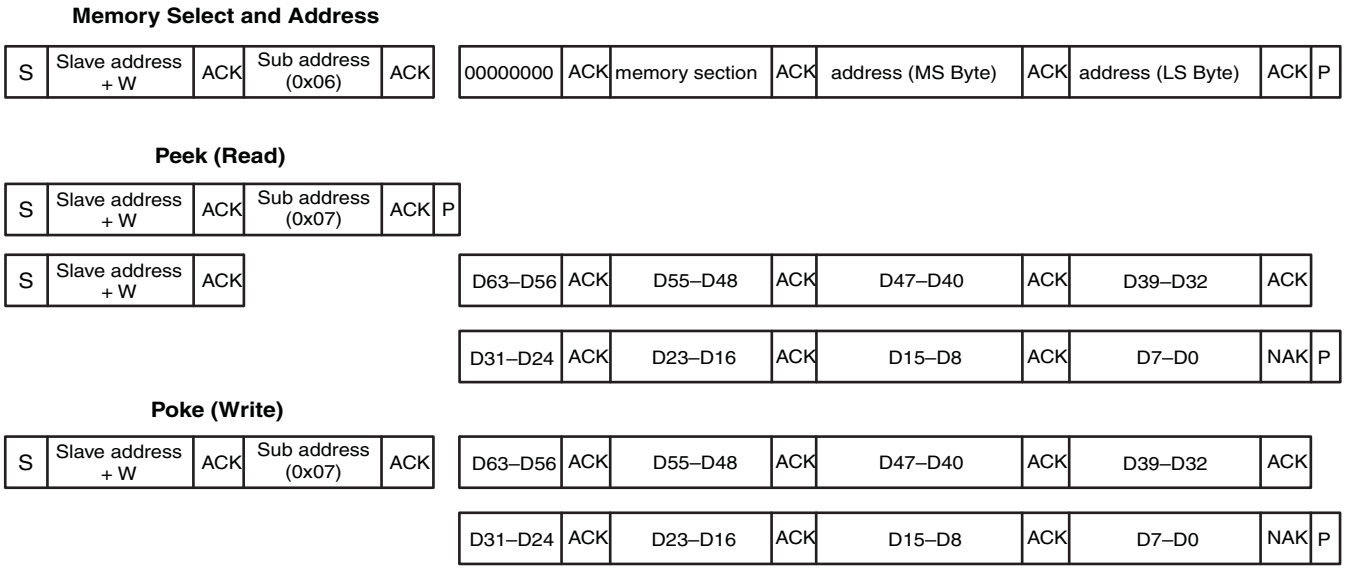


Figure 40. I^2C Transaction for PEEK/POKE

Mute Control Register (0x09)

Table 34. Mute Control Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	0	0	0	0	0	0	0	0	Unused
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0	0	Unused
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0	0							Unused
			AMX1	AMX0					Analog MUX out (LINEOUT1)
					SD2	SD2			SDOUT2/SPDIFOUT
							SD1	SD1	SDOUT1
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	DAC1	DAC1							DAC1
			DAC2	DAC2					DAC2
					DAC3	DAC3			DAC3
							DIT	DIT	DIT

Table 35. Mute[(1)]

MUTE	MUTE[1]	MUTE[0]
Hardware controlled	*0*	*0*
Force mute off	0	1
Force mute on	1	0

(1) Default values are shown in italics.

GPIO Control Register (0x0c)

Table 36. GPIO Control Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	WDE								Watchdog timer
		0	0	0					Unused
					IO2				GPIO2 input/output value
						IO1			GPIO1 input/output value
							DIR2		GPIO2 direction
								DIR1	GPIO1 direction
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	x	x	x	x	x	x	x	x	GPIOMICROCOUNT MSB
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	x	x	x	x	x	x	x	x	GPIOMICROCOUNT LSB
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	y	y	y	y	y	y	y	y	GPIO_Sampling_Interval

GPIOMICROCOUNT sets the number of micro clock cycles for Timer 0 interrupt. In Timer 0 interrupt service routine, the watchdog timer is reset if it is enabled. The default value for this counter is 0x5820, which corresponds to a period 1.25 ms.

Table 37. Watchdog Timer Enable[(1)]

WATCHDOG TIMER	WDE
Enable	0

(1) Default values are shown in italics.

**Table 37.
Watchdog Timer Enable(1)
(continued)**

WATCHDOG TIMER	WDE
Disable	*1*

Table 38. GPIO Direction(1)

GPIOx DIRECTION	DIRx
Output	0
Input	1

(1) Default values are shown in italics.

Powerdown Control Register (0x10)

Table 39. Powerdown Control Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	0	0	0	0	0	0	0	0	Unused
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0	0	Unused
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0	0	0	0	0	0	0	0	Unused
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	DIT								DIT reset
		DAC 3							DAC3
			DAC 2						DAC2
				DAC 1					DAC1
					ADC				AMUX + AAF + ADC
						0			Unused
							0		Unused
								AMX1	AMUX1 + LineAmp1

Table 40. Powerdown(1)

POWERDOWN	PD
Powerdown and disable	*0*
Powerup and enable	1

(1) Default values are shown in italics.

A-MUX Control Register(0x12)

Table 41. A-MUX Control Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	x	x	x	x	1	1	1	1	Reserved
	x	x	x	x	1	1	1	0	Reserved
	x	x	x	x	1	1	0	1	Reserved
	x	x	x	x	1	1	0	0	Reserved
	x	x	x	x	1	0	1	1	DAC
	x	x	x	x	1	0	1	0	Analog MUX line 10 select
	x	x	x	x	1	0	0	1	Analog MUX line 9 select
	x	x	x	x	1	0	0	0	Analog MUX line 8 select
	x	x	x	x	0	1	1	1	Analog MUX line 7 select
	x	x	x	x	0	1	1	0	Analog MUX line 6 select
	x	x	x	x	0	1	0	1	Analog MUX line 5 select
	x	x	x	x	0	1	0	0	Analog MUX line 4 select
	x	x	x	x	0	0	1	1	Analog MUX line 3 select
	x	x	x	x	0	0	1	0	Analog MUX line 2 select
	x	x	x	x	0	0	0	1	Analog MUX line 1 select
	0	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*MUTE*
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0	0	Unused
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0	0	0	0	0	0	0	0	Unused
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	x	x	x	x	1	1	1	1	Reserved
	x	x	x	x	1	1	1	0	Reserved
	x	x	x	x	1	1	0	1	Reserved
	x	x	x	x	1	1	0	0	Reserved
	x	x	x	x	1	0	1	1	DAC
	x	x	x	x	1	0	1	0	Analog MUX line 10 select
	x	x	x	x	1	0	0	1	Analog MUX line 9 select
	x	x	x	x	1	0	0	0	Analog MUX line 8 select
	x	x	x	x	0	1	1	1	Analog MUX line 7 select
	x	x	x	x	0	1	1	0	Analog MUX line 6 select
	x	x	x	x	0	1	0	1	Analog MUX line 5 select
	x	x	x	x	0	1	0	0	Analog MUX line 4 select
	x	x	x	x	0	0	1	1	Analog MUX line 3 select
	x	x	x	x	0	0	1	0	Analog MUX line 2 select
	x	x	x	x	0	0	0	1	Analog MUX line 1 select
	0	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*MUTE*

SPDIF Control Register (0x16)

Table 42. SPDIF Control Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	CP								Copyright flag
		EMP							Pre-emphasis flag
			CLKAC	CLKAC					Clock accuracy
			b28	b29					
					WL3	WL2	WL1	WL0	Sample word length
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	SR	SR							Sampling rate
	b24	b25	0	0	0	0	0	0	
			VL						Left-channel validity flag
				VR					Right-channel validity flag
					SRC#	SRC#	SRC#	SRC#	Source channel number
					b19	b18	b17	b16	
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	Cat	Cat	Cat	Cat	Cat	Cat	Cat		Category code
	b8	b9	b10	b11	b12	b13	b14	0	
								L	Generation status
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	0	0	0	0	0	0			Unused
							MUX1	MUX0	SPDIF MUX

Table 43. Copyright Flag(1)

COPYRIGHT FLAG	CP
Copy prohibited	*0*
Copy permitted	1

(1) Default values are shown in italics.

Table 44. Pre-Emphasis Flag(1)

PRE-EMPHASIS FLAG	EMP
No pre-emphasis	*0*
50/15 µs pre-emphasis	1

(1) Default values are shown in italics.

Table 45. Sample Word Length

SAMPLE WORD LENGTH	WLx
24-bit sample word length	0

Table 46. Sampling Rate

SAMPLING RATE	b24	b25
48 kHz	0	1

Table 47. Validity Flag(1)

VALIDITY FLAG	Vx
Valid	*0*
Not valid	1

(1) Default values are shown in italics.

Table 48. Channel Source Number

CHANNEL SOURCE NUMBER	b19	b18	b17	b16
Channel 2	0	0	1	0

Table 49. Category Code

CATEGORY CODE	b8	b9	b10	b11	b12	b13	b14
Digital sound processor	0	1	0	1	0	1	0

Table 50. Generation Status

GENERATION STATUS	Vx
Gen 1 or higher	0
Original	1

Table 51. SDOUT/SPDIF MUX(1)

SDOUT/SPDIF MUX	MUX1	MUX2
SDOUT2	*0*	*0*
SPDIF Tx	0	1
SPDIF In	1	–

(1) Default values are shown in italics.

DC Dither Register (0x1d)

Table 52. DC Dither Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	0	0	0	0	0	0	0	0	Unused
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0	0	Unused
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0	0	0	0	0	0	0	0	Unused
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	0	0	0	0	0	0	0		Unused
								ON	DC dither enable

Table 53. DC Dither Enable(1)

DC DITHER ENABLE	ON
Disable	0
Enable	*1*

(1) Default values are shown in italics.

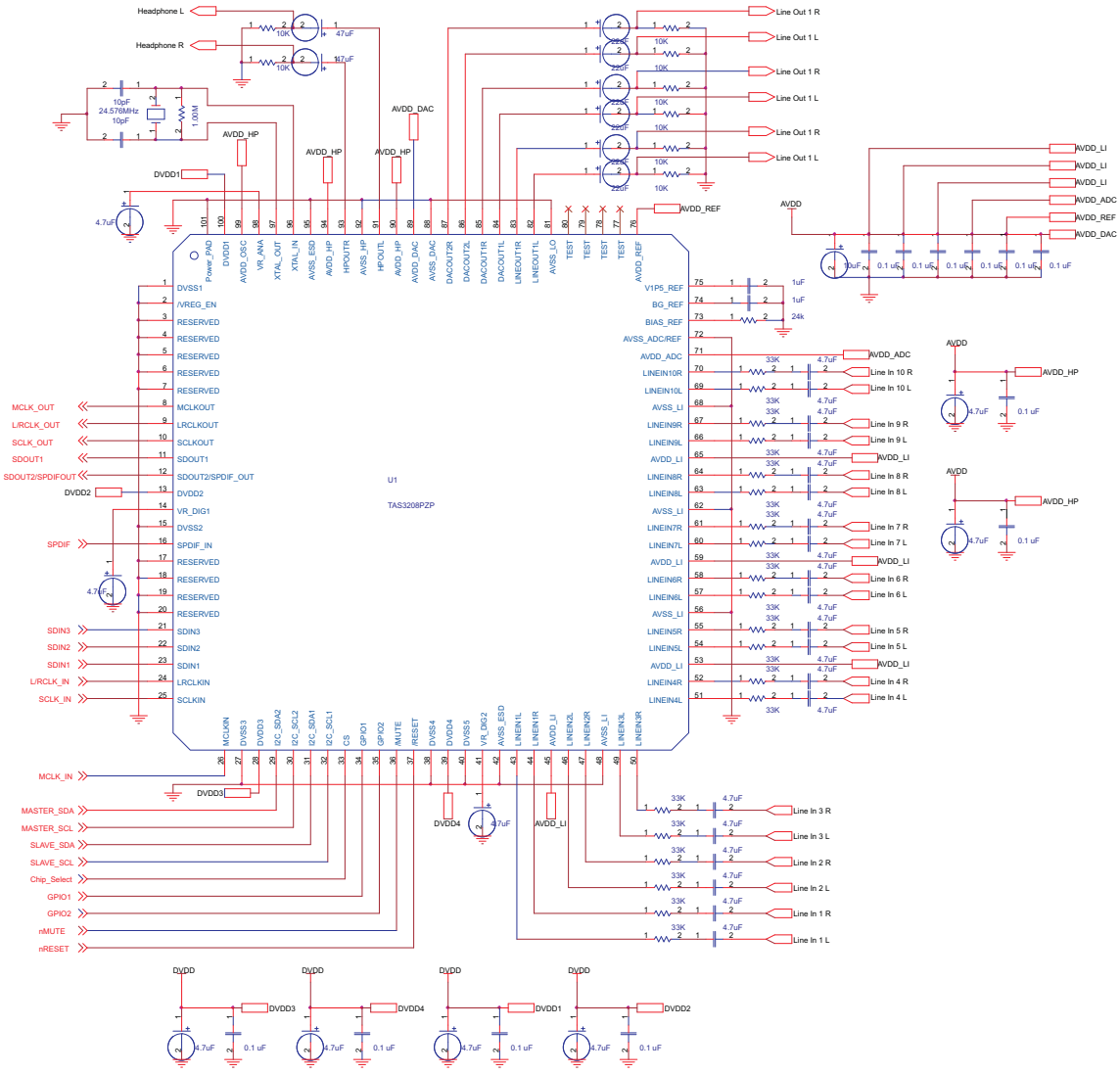
DSP Program Start Address Register (0x1e)

The DSP instruction execution loops each Fs cycle. At the beginning of the Fs cycle, the DSP instruction pointer is set to the starting address specified in the 12 LSBs. The maximum address is the end address of DSP instruction address 3327.

Table 54. DSP Program Start Address Register

BIT	31	30	29	28	27	26	25	24	DESCRIPTION
	0	0	0	0	0	0	0	0	Unused
BIT	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	
	0	0	0	0	0	0	0	0	Unused
BIT	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	
	0	0	0	0	x	x	x	x	Starting address MSB
BIT	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	
	x	x	x	x	x	x	x	x	Starting address LSB

APPLICATION INFORMATION



PACKAGING INFORMATION

Orderable part number	Status (1)	Material type (2)	Package | Pins	Package qty | Carrier	RoHS (3)	Lead finish/ Ball material (4)	MSL rating/ Peak reflow (5)	Op temp (°C)	Part marking (6)
TAS3208IPZP	Active	Production	HTQFP (PZP) | 100	90 | JEDEC TRAY (10+1)	Yes	NIPDAU	Level-3-260C-168 HR	-40 to 85	TAS3208IPZP
TAS3208IPZP.A	Active	Production	HTQFP (PZP) | 100	90 | JEDEC TRAY (10+1)	Yes	NIPDAU	Level-3-260C-168 HR	-40 to 85	TAS3208IPZP
TAS3208PZP	Obsolete	Production	HTQFP (PZP) | 100	-	-	Call TI	Call TI	-20 to 70	TAS3208PZP

(1) **Status:** For more details on status, see our product life cycle.

(2) **Material type:** When designated, preproduction parts are prototypes/experimental devices, and are not yet approved or released for full production. Testing and final process, including without limitation quality assurance, reliability performance testing, and/or process qualification, may not yet be complete, and this item is subject to further changes or possible discontinuation. If available for ordering, purchases will be subject to an additional waiver at checkout, and are intended for early internal evaluation purposes only. These items are sold without warranties of any kind.

(3) **RoHS values:** Yes, No, RoHS Exempt. See the TI RoHS Statement for additional information and value definition.

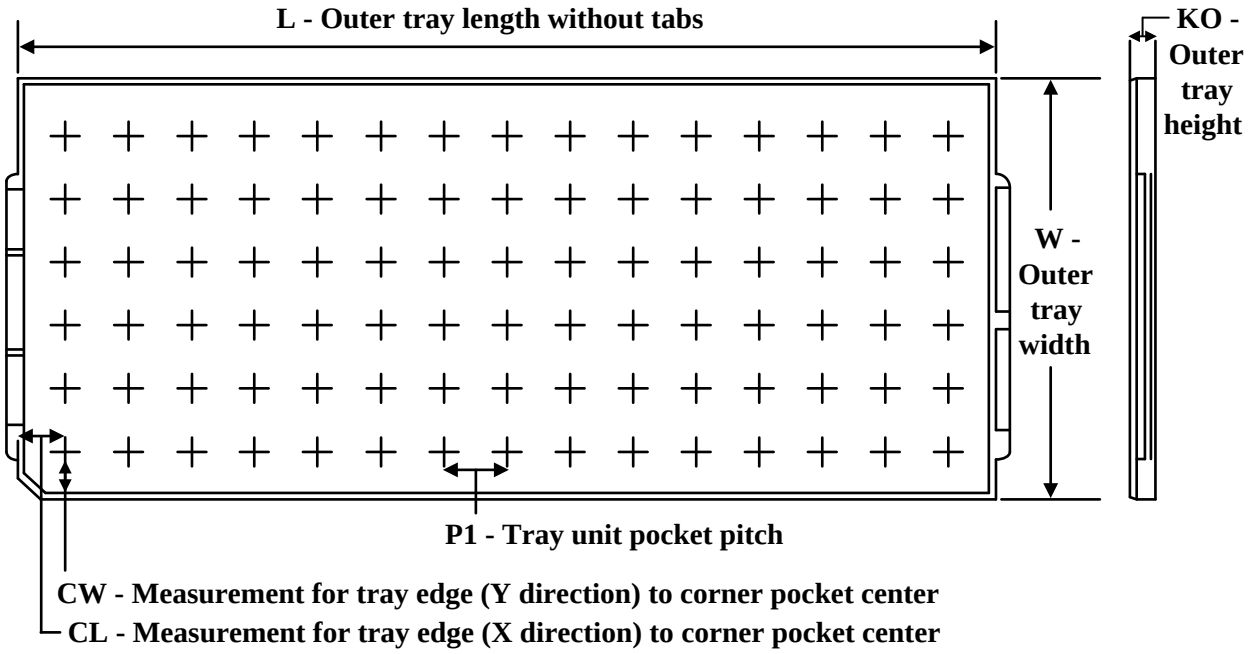
(4) **Lead finish/Ball material:** Parts may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

(5) **MSL rating/Peak reflow:** The moisture sensitivity level ratings and peak solder (reflow) temperatures. In the event that a part has multiple moisture sensitivity ratings, only the lowest level per JEDEC standards is shown. Refer to the shipping label for the actual reflow temperature that will be used to mount the part to the printed circuit board.

(6) **Part marking:** There may be an additional marking, which relates to the logo, the lot trace code information, or the environmental category of the part.

Multiple part markings will be inside parentheses. Only one part marking contained in parentheses and separated by a "~" will appear on a part. If a line is indented then it is a continuation of the previous line and the two combined represent the entire part marking for that device.

TRAY



Chamfer on Tray corner indicates Pin 1 orientation of packed units.

*All dimensions are nominal

Device	Package Name	Package Type	Pins	SPQ	Unit array matrix	Max temperature (°C)	L (mm)	W (mm)	K0 (µm)	P1 (mm)	CL (mm)	CW (mm)
TAS3208IPZP	PZP	HTQFP	100	90	6 X 15	150	315	135.9	7620	20.3	15.4	15.45
TAS3208IPZP.A	PZP	HTQFP	100	90	6 X 15	150	315	135.9	7620	20.3	15.4	15.45

GENERIC PACKAGE VIEW

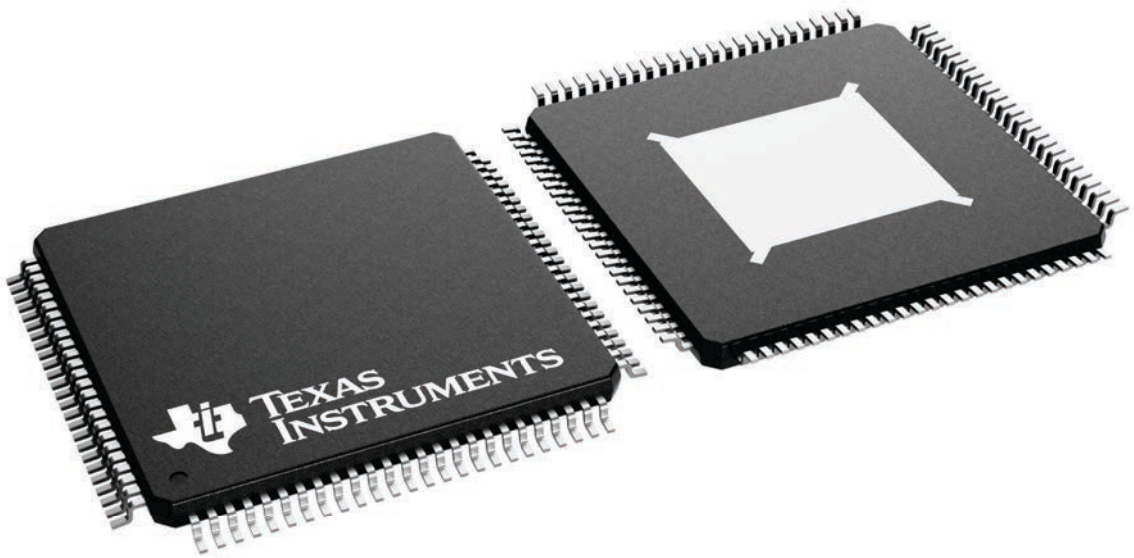
PZP 100

PowerPAD ™ TQFP - 1.2 mm max height

14 x 14 mm Pkg Body, 0.5 mm pitch
16 x 16 mm Pkg Area

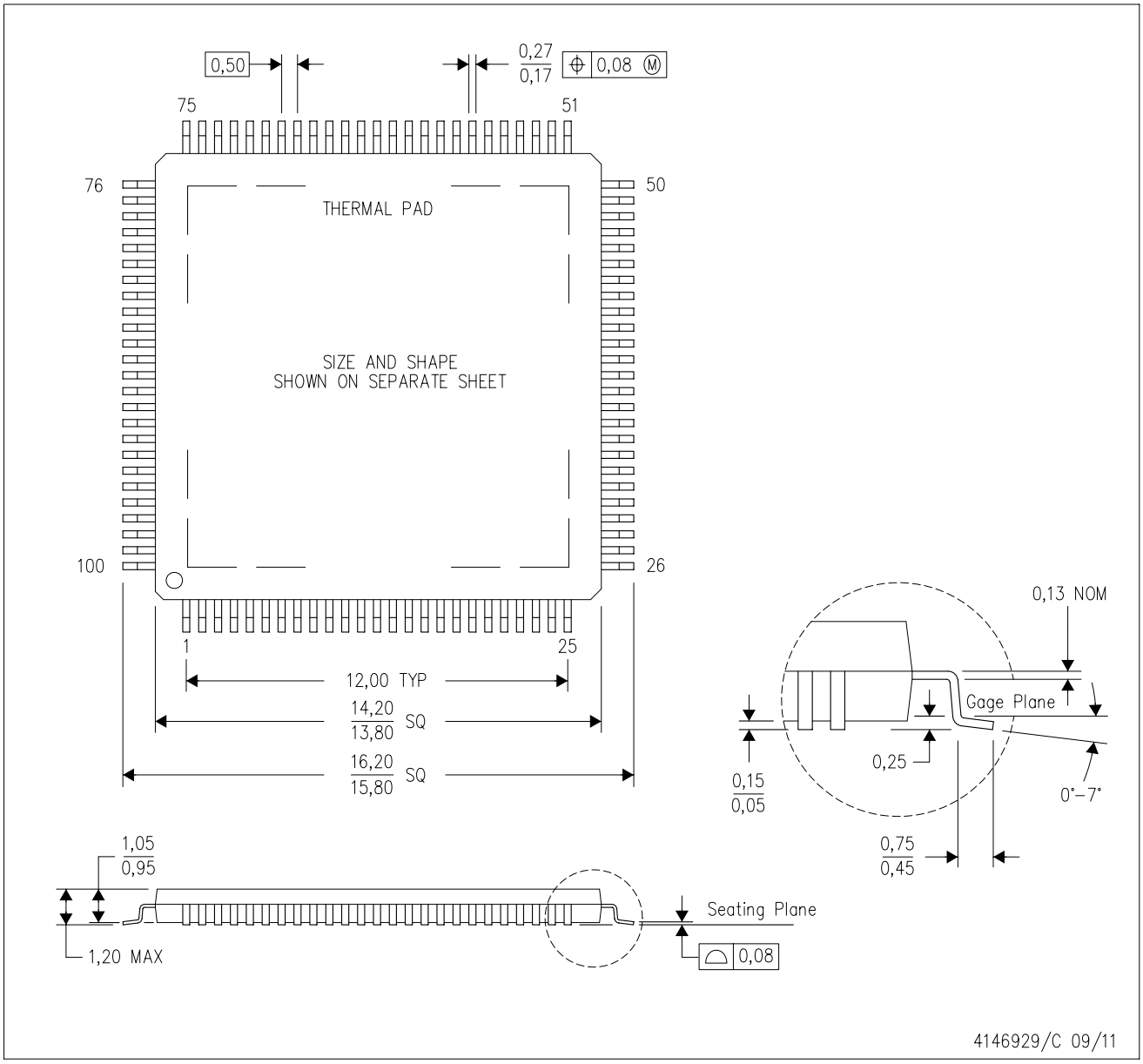
PLASTIC QUAD FLATPACK

This image is a representation of the package family, actual package may vary.
Refer to the product data sheet for package details.



4224739/B

PZP (S–PQFP–G100) PowerPAD™ PLASTIC QUAD FLATPACK



NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Body dimensions do not include mold flash or protrusion
D. This package is designed to be soldered to a thermal pad on the board. Refer to Technical Brief, PowerPad Thermally Enhanced Package, Texas Instruments Literature No. SLMA002 for information regarding recommended board layout. This document is available at www.ti.com <http://www.ti.com>.
E. See the additional figure in the Product Data Sheet for details regarding the exposed thermal pad features and dimensions.
F. Falls within JEDEC MS–026

PowerPAD is a trademark of Texas Instruments.

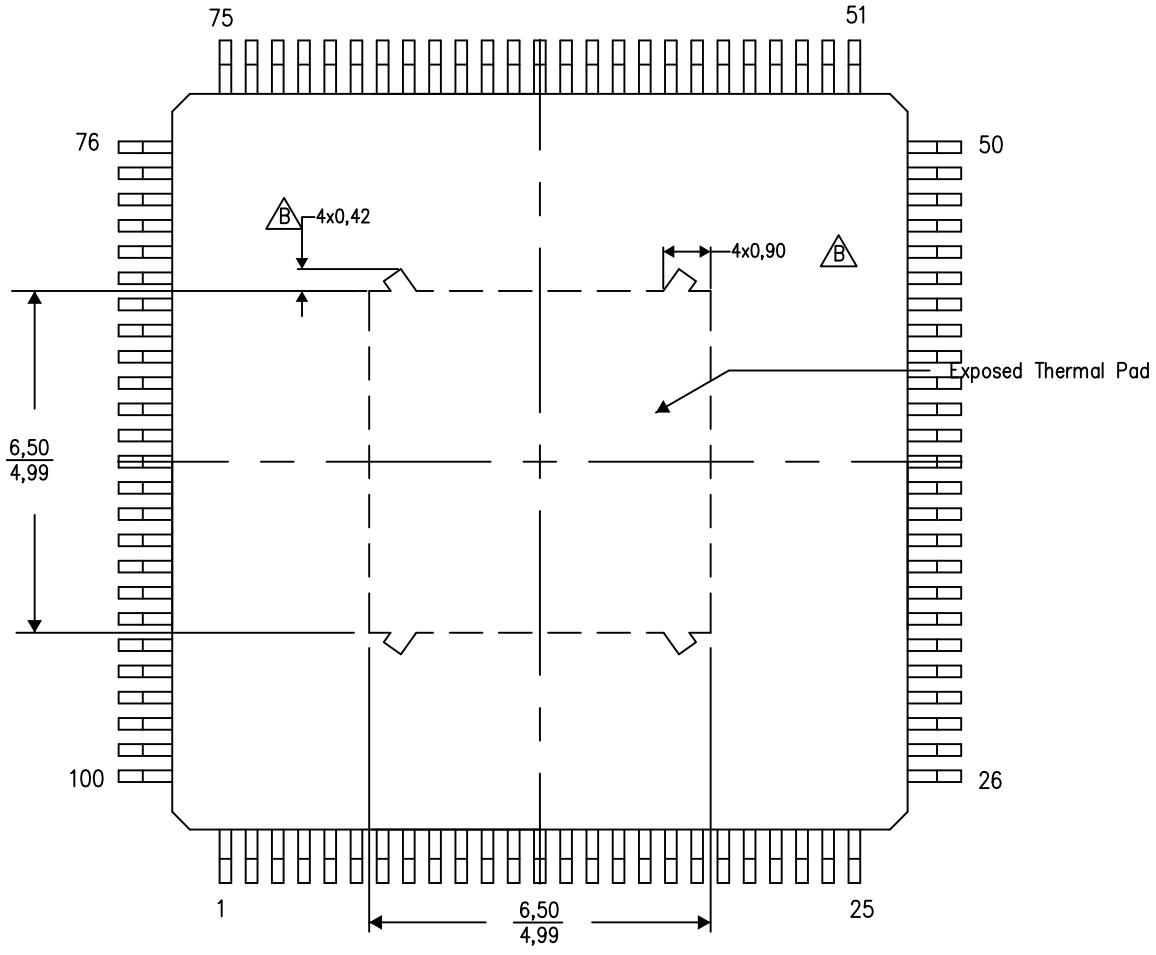
PZP (S–PQFP–G100) PowerPAD™ PLASTIC QUAD FLATPACK

THERMAL INFORMATION

This PowerPAD™ package incorporates an exposed thermal pad that is designed to be attached to a printed circuit board (PCB). The thermal pad must be soldered directly to the PCB. After soldering, the PCB can be used as a heatsink. In addition, through the use of thermal vias, the thermal pad can be attached directly to the appropriate copper plane shown in the electrical schematic for the device, or alternatively, can be attached to a special heatsink structure designed into the PCB. This design optimizes the heat transfer from the integrated circuit (IC).

For additional information on the PowerPAD package and how to take advantage of its heat dissipating abilities, refer to Technical Brief, PowerPAD Thermally Enhanced Package, Texas Instruments Literature No. SLMA002 and Application Brief, PowerPAD Made Easy, Texas Instruments Literature No. SLMA004. Both documents are available at www.ti.com.

The exposed thermal pad dimensions for this package are shown in the following illustration.



Top View

Exposed Thermal Pad Dimensions

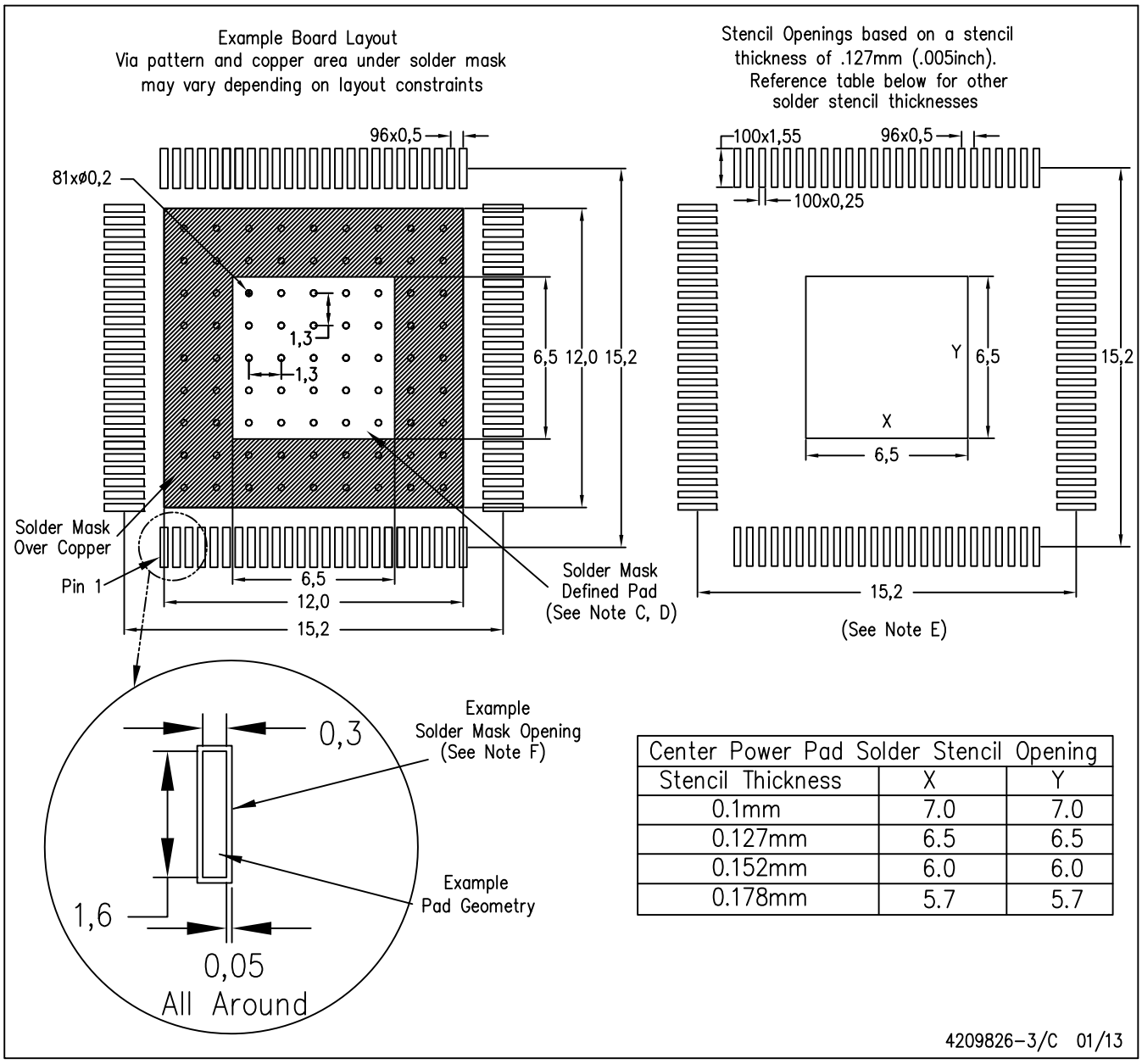
4206333–5/L 05/14

NOTE: A. All linear dimensions are in millimeters

B Tie strap features may not be present.

PowerPAD is a trademark of Texas Instruments

PZP (S–PQFP–G100) PowerPAD™ PLASTIC QUAD FLATPACK



Center Power Pad Solder Stencil Opening		
Stencil Thickness	X	Y
0.1mm	7.0	7.0
0.127mm	6.5	6.5
0.152mm	6.0	6.0
0.178mm	5.7	5.7

NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Customers should place a note on the circuit board fabrication drawing not to alter the center solder mask defined pad.
D. This package is designed to be soldered to a thermal pad on the board. Refer to Technical Brief, PowerPad Thermally Enhanced Package, Texas Instruments Literature No. SLMA002, SLMA004, and also the Product Data Sheets for specific thermal information, via requirements, and recommended board layout. These documents are available at www.ti.com <http://www.ti.com>. Publication IPC–7351 is recommended for alternate designs.
E. Laser cutting apertures with trapezoidal walls and also rounding corners will offer better paste release. Customers should contact their board assembly site for stencil design recommendations. Example stencil design based on a 50% volumetric metal load solder paste. Refer to IPC–7525 for other stencil recommendations.
F. Customers should contact their board fabrication site for solder mask tolerances between and around signal pads.

PowerPAD is a trademark of Texas Instruments.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265