Data Sheet

ADAU1797

High-Performance Audio Codec with Integrated HiFi 3z and FastDSP Cores

FEATURES

- FastDSP™ audio processing engine
  - Low latency at sample rates up to 768kHz
  - Supports up to 128 instructions
- Tensilica® HiFi 3z DSP with TIE accelerators
  - Quad MAC/cycle and 1024kB total memory
  - 2-way SIMD IEEE floating point multiplier
  - Supports both 50MHz and 200MHz modes
- 5µs group delay (analog input to amp output)
  - 768kHz sample rate (FastDSP bypass mode)
- Three low power 24-bit ADC record channels
  - 104dB SNR, -91dB THD+N, 0.6mW (PGA on)
  - Flexible differential or single-ended inputs
- Ten digital microphone (DMIC) input channels
  - Two clock outputs (384kHz to 6.144MHz)
- Mono low power 24-bit DAC playback channel
  - 113dB SNR, -93dB THD+N, 1.4mW P_Q
  - High-efficiency, low-noise, Class-D amplifier
- Two high-performance PDM output channels
  - Supports 3.072MHz or 6.144MHz clock rates
- Two 32-bit I2S/TDM serial audio data ports
  - Sync clock frequencies from 8kHz to 768kHz
  - Two full-duplex, 4-channel ASRCs
- Supports low power, single-supply mode (1.8V)
  - Integrated LDO or switch-cap regulator
  - Digital I/O supports 1.2V or 1.8V logic levels
- Digital Control and Communication interfaces
  - I^2C/SPI control and I^3C combined interfaces
  - Master QSPI, UART, and JTAG interfaces
  - Supports self-boot from QSPI Flash/EEPROM
- Multi-purpose I/O pins for GPIO / IRQ support

APPLICATIONS

- True wireless stereo (TWS) ANC headphones
- Over-ear stereo ANC headphones
- VR and AR headsets and wearable devices
- Hearing assist and PSAP devices
- Soundbar and smart speaker systems
- Gaming devices and tablets

GENERAL DESCRIPTION

The device is a low power, high-performance audio codec that provides three analog input channels, ten DMIC input channels, two PDM output channels, and one high-efficiency Class-D amplifier output channel.

The device features a low power HiFi 3z audio DSP core and a low-latency FastDSP core. The audio DSP cores paired with high-fidelity audio data converters are ideal for applications like noise cancellation, transparency, personal sound amplification, and voice processing.

When operating in low-power mode, the DSP cores are optimized for small form factor applications such as true wireless stereo (TWS) headphones. In this mode, the device delivers the right level of processing power while still minimizing power consumption to extend play time.

The device has the flexibility to also support applications requiring additional processing capability such as over-ear headphones, VR and AR headsets, wearables, hearing assist, and PSAP devices. In high-performance mode, the HiFi 3z core is boosted from 50Mhz to 200MHz, and the FastDSP supports double the number of instructions (up from 64 to 128). This increased processing capability can either be used to offload cycles from the host processor or enable a lower-cost host processor without requiring an additional external audio DSP or MCU.

The device supports a -40°C to +85°C temperature range and is available in a space-saving 77-ball WLCSP package (0.4mm pitch, 3.24mm × 4.83mm)

One Analog Way, Wilmington, MA 01887-2356, U.S.A. | Tel: 781.329.4700

FUNCTIONAL BLOCK DIAGRAM

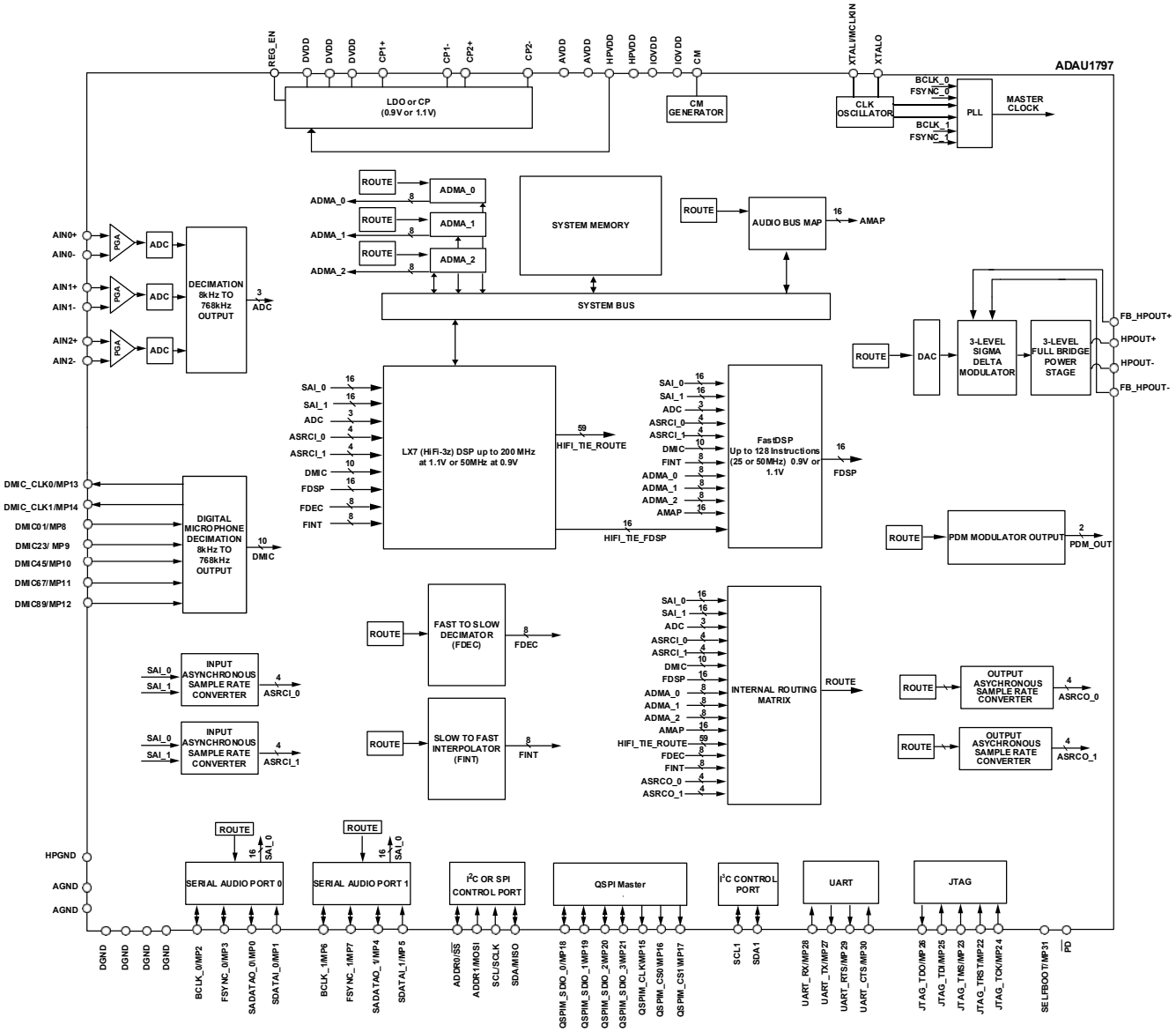

Figure 1. Top Level Block Diagram

REVISION HISTORY

9/2023—Revision 0: Initial Version

SPECIFICATIONS

Table 1. Electrical Characteristics

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
POWER SUPPLY VOLTAGE RANGE							
AVDD Voltage	V_{AVDD}			1.7	1.8	1.98	V
AVDD Undervoltage Trip Point					1.55		V
HPVDD Voltage	V_{HPVDD}			1.7	1.8	1.98	V
HPVDD Undervoltage Trip Point					1.55		V
DVDD Voltage	V_{DVDD}	Integrated supply	HIFI_SPEED = 0	0.85	0.9	0.99	V
		External supply	HIFI_SPEED = 0	0.85	0.9	1.21	
		Integrated or external supply	HIFI_SPEED = 1	1.05	1.1	1.21	
IOVDD Voltage	V_{IOVDD}			1.1	1.8	1.98	V
ANALOG-TO-DIGITAL CONVERTERS (ADCs)							
ADC Resolution					24		Bits
Digital Volume Step					0.375		dB
Digital Volume Range				-71		+24	dB
Digital Volume Ramp Rate					4.5		dB/ms
ANALOG INPUT RESISTANCE (AINPx/AINNx R_{IN})							
Input Disabled					≥1		MΩ
Direct ADC Input Mode					21		kΩ
Programmable Gain Amplifier Input Mode		Differential input mode	-9dB attenuation		33.7		kΩ
			0dB gain		22.5		
			6dB gain		15		
			28.5dB gain		1.5		
		Single-ended or pseudo-diff. input mode	-9dB attenuation		26.5		
			0dB gain		15		
			6dB gain		9.2		
			28.5dB gain		0.9		

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
SINGLE-ENDED OR PSEUDO-DIFFERENTIAL LINE INPUT MODE (DIRECT TO ADC INPUT)							
Full-Scale Input Voltage		Input level for a 0dBFS output			0.64		V_{RMS}
					1.8		V_{P-P}
Dynamic Range[1]		A-weighted filter, f_{IN} = 1kHz, -60dBFS output	Normal operation mode (default)		100		dB
Dynamic Range[1]		A-weighted filter, f_{IN} = 1kHz, -60dBFS output	Extreme power-saving mode		96		dB
		Unweighted filter, f_{IN} = 1kHz, -60dBFS output	Normal operation mode (default)		97		
			Extreme power-saving mode		93		
Signal-to-Noise Ratio (SNR)[2]		Normal operation mode	A-weighted filter		100		dB
			Unweighted filter		97		
THD+N Level[3]		f_{IN} = 1kHz, -1dBFS output, normal operation mode			-87		dBFS
Interchannel Gain Mismatch					0.04		dB
Offset Error					±0.1		mV
Gain Error					±0.2		dB
Interchannel Isolation		C_{CM} = 10µF			100		dB
Power Supply Rejection Ratio (PSRR)		C_{CM} = 10µF, 100mV_{P-P} signal	f_{IN} = 1kHz		70		dB
			f_{IN} = 10kHz		50		
DIFFERENTIAL LINE INPUT MODE (DIRECT TO ADC INPUT)							
Full-Scale Input Voltage		Input level for a 0dBFS output			1.08		V_{RMS}
					3.05		V_{P-P}
Dynamic Range[1]		A-weighted filter, f_{IN} = 1kHz, -60dBFS output	Normal operation mode (default)		106		dB
			Extreme power-saving mode		100		

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
		Unweighted filter, f_{IN} = 1kHz, -60dBFS output	Normal operation mode (default)		103		
			Extreme power-saving mode		97		
Signal-to-Noise Ratio (SNR)[2]		Normal operation mode	A-weighted filter		106		dB
			Unweighted filter		103		
THD+N Level[3]		f_{IN} = 1kHz, -1dBFS output, normal operation mode			-93		dBFS
CMRR					60		dB
Interchannel Gain Mismatch					0.04		dB
Offset Error					±0.1		mV
Gain Error					±0.2		dB
Interchannel Isolation		C_{CM} = 10µF			100		dB
Power Supply Rejection Ratio (PSRR)		C_{CM} = 10µF, 100mV_{P-P} signal	f_{IN} = 1kHz		70		dB
			f_{IN} = 10kHz		50		
SINGLE-ENDED OR PSEUDO-DIFFERENTIAL PGA INPUT MODE (PGA ENABLED)							
Full-Scale Input Voltage		Input level for a 0dBFS output			0.54		V_{RMS}
					1.53		V_{P-P}
Dynamic Range[1]		A-weighted filter, f_{IN} = 1kHz, -60dBFS output, +6dB PGA gain	Normal operation mode (default)		99		dB
			Extreme power-saving mode		98		
		Unweighted filter, f_{IN} = 1kHz, -60dBFS output, +6dB PGA gain	Normal operation mode (default)		96		
			Extreme power-saving mode		94		
Signal-to-Noise Ratio (SNR)[2]		+6dB PGA gain, normal operation mode	A-weighted filter		98		dB
			Unweighted filter		95		
THD+N Level[3]		f_{IN} = 1kHz, -1dBFS output, +6dB PGA gain, normal operation mode			-90		dBFS

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
PGA Gain Range				-9		+29.25	dB
PGA Gain Step Size					0.75		dB
PGA Gain Variation with 0dB Setting					0.05		dB
PGA Gain Variation with +24dB Setting					0.15		dB
Interchannel Gain Mismatch					0.005		dB
Offset Error					±0.1		mV
Gain Error					±0.2		dB
Interchannel Isolation		C_{CM} = 10µF			100		dB
Power Supply Rejection Ratio (PSRR)		C_{CM} = 10µF, 100mV_{P-P} signal, f_{IN} = 1kHz			70		dB
DIFFERENTIAL PGA INPUT MODE (PGA ENABLED)							
Full-Scale Input Voltage		Input level for a 0dBFS output			1.08		V_{RMS}
					3.05		V_{P-P}
Dynamic Range[1]		A-weighted filter, f_{IN} = 1kHz, -60dBFS output, 0dB PGA gain	Normal operation mode		104		dB
		A-weight(RMS)ed filter, f_{IN} = 1kHz, -60dBFS output, 0dB PGA gain	Extreme power-saving mode		100		
		Unweighted filter, f_{IN} = 1kHz, -60dBFS output, 0dB PGA gain	Normal operation mode		100		
			Extreme power-saving mode		97		
Signal-to-Noise Ratio (SNR)[2]		0dB PGA gain, normal operation mode	A-weighted filter		103		dB
			Unweighted filter		100		
THD+N Level[3]		f_{IN} = 1kHz, -1dBFS output, 0dB PGA gain, normal operation mode			-93		dBFS
PGA Gain Range				-6		+30	dB
PGA Gain Step Size					0.75		dB

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS	MIN	TYP	MAX	UNITS
PGA Gain Variation With 0dB Setting				0.05		dB
PGA Gain Variation With +24dB Setting				0.15		dB
CMRR		1kHz input frequency		60		dB
Interchannel Gain Mismatch				0.005		dB
Offset Error				±0.1		mV
Gain Error				±0.2		dB
Interchannel Isolation		C_{CM} = 10µF		100		dB
Power Supply Rejection Ratio (PSRR)		C_{CM} = 10µF, 100mV_{P-P} signal, f_{IN} = 1kHz		70		dB
DIGITAL-TO-ANALOG CONVERTER (DAC)						
DAC Resolution				24		Bits
Digital Volume Step				0.375		dB
Digital Volume Range			-71		+24	dB
Digital Volume Ramp Rate				4.5		dB/ms
DIFFERENTIAL HEADPHONE OUTPUT						
Full-Scale Output Voltage		0dBFS input to analog output channel, 16Ω load		1.15		V_{RMS}
Dynamic Range[1]		A-weighted filter, f_{IN} = 1kHz, -60dBFS input		113		dB
		Unweighted filter, f_{IN} = 1kHz, -60dBFS input		110		
Signal-to-Noise Ratio (SNR)[2]		A-weighted filter		113		dB
		Unweighted filter		110		
Output Noise		A-weighted filter, zero code digital input		2.57		µV_{RMS}
THD+N Level[3]		-2dBFS, 16Ω load		-90		dBV
THD+N Ratio		32Ω load, P_{OUT} = 1mW		-83		dB
		32Ω load, P_{OUT} = 30mW		-90		
		24Ω load, P_{OUT} = 40mW		-90		
		16Ω load, P_{OUT} = 60mW		-90		

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
Headphone Output Power		HPVDD = 1.8V, <0.1% THD+N ratio	32Ω load		40		mW
		HPVDD = 1.8V, <0.1% THD+N ratio	24Ω load		54		
			16Ω load		80		
Gain Error					±2.5		%
DC Offset					±0.1		mV
Pop-Click Level		A-weighted peak transient voltage			0.5		mV_P
Peak Output Current						350	mA
Minimum Load Resistance				6			Ω
Minimum Load Inductance				5			µH
Maximum Load Capacitance		From HPOUTP or HPOUTN to GND or HPVDD				470	pF
Power Efficiency		55mW into a 16Ω load			90		%
PSRR		100mV_{P-P} signal	f_{IN} = 1kHz		89		dB
			f_{IN} = 10kHz		75		
COMMON MODE REFERENCE							
Output					0.85		V
Source Impedance					5		kΩ
PHASED-LOCKED LOOP (PLL)							
Input Frequency		After input prescale		0.9		2.1	MHz
Output Frequency				195.5	196.608	198	MHz
Fractional Limits		Fractional mode, fraction portion (N/M ratio)		0.1		0.9	
Integer Limits		Fractional mode, integer portion		2		256	
Input Pre-Divide				1		32	
Lock Time		1.024MHz input				1	ms
PLL Bypass XTAL Frequency					49.152		MHz
INTEGRATED LDO REGULATOR							
Line Regulation					1.1		mV/V
Load Regulation					0.3		mV/mA

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
INTEGRATED SWITCHED-CAP REGULATOR							
Line Regulation		V_{HPVDD} = 1.8V, see *Figure 47* for full HPVDD range			5		mV/V
Load Regulation					0.3		mV/mA
CRYSTAL AMPLIFIER							
Jitter					270	500	ps
Frequency Range				0.9		50	MHz
Load Capacitance						20	pF
DIGITAL INPUT AND OUTPUT							
Input Voltage High	V_{IH}			0.7 × V_{IOVDD}			V
Input Voltage Low	V_{IL}					0.3 × V_{IOVDD}	V
Input Leakage High	I_{IH}	V_{IOVDD} = 1.8V	V_{IH} = 1.1V			10	µA
Input Leakage Low	I_{IL}	V_{IOVDD} = 1.8V	V_{IL} = 0.45V			10	µA
Output Voltage High	V_{OH}			0.71 × V_{IOVDD}	0.83 × V_{IOVDD}		V
Output High Current Drive Strength	I_{OH}	Low output drive strength			1		mA
		High output drive strength			3		
Output Voltage Low	V_{OL}				0.1 × V_{IOVDD}	0.3 × V_{IOVDD}	V
Output Low Current Drive Strength	I_{OL}	Low output drive strength			1		mA
		High output drive strength			3		
Input Capacitance						5	pF
SERIAL AUDIO DATA PORT DIGITAL INTERFACE CURRENT							
Digital Current (IOVDD)		Single serial audio data port disabled (SPTx), crystal oscillator enabled (24.576MHz) V_{IOVDD} = 1.8V			0.422		mA
		Single serial audio data port enabled (SPTx), crystal oscillator enabled (24.576MHz),	Slave mode, f_S = 48kHz, f_{BCLK} = 3.072MHz		0.475		
			Slave mode, f_S = 192kHz, f_{BCLK} = 12.288MHz		0.481		

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
		25pF line loads, V_{IOVDD} = 1.8V	Master mode, f_S = 48kHz, f_{BCLK} = 3.072MHz		0.697		
			Master mode, f_S = 192kHz, f_{BCLK} = 12.288MHz		1.336		
ADC INPUT TO DAC OUTPUT SIGNAL PATH							
Pass-Band Ripple		DC to 20kHz Bandwidth, f_S = 192kHz, ADCxx_FCOMP = 1, DAC_FCOMP = 1				±0.02	dB
Group Delay[4]		f_S = 192kHz			12.9		µs
		f_S = 384kHz			7.5		
		f_S = 768kHz			5		
ASYNCHRONOUS SAMPLE RATE CONVERTERS							
Pass-Band Frequency		f_{LRCLK} < 63kHz				0.475 x f_S	kHz
		63kHz ≤ f_{LRCLK} < 112kHz				0.4286 x f_S	
		112kHz ≤ f_{LRCLK}				0.4286 x f_S	
Audio-Band Ripple		20Hz to 20kHz bandwidth		-0.1		+0.1	dB
Sample Rate Frequency Range		Input and output ASRCs		7		224	kHz
Dynamic Range[1]		ASRCIx_LPM or ASRCOx_LPM = 0			130		dB
		ASRCIx_LPM or ASRCOx_LPM = 1			130		
		ASRCIx_LPM_II or ASRCOx_LPM_II = 1			130		
THD+N Level[3, 6]		ASRCIx_LPM or ASRCOx_LPM = 0			-130	-120	dBFS
		ASRCIx_LPM or ASRCOx_LPM = 1			-120	-110	
		ASRCIx_LPM_II or ASRCOx_LPM_II = 1			-115	-90	
Startup Lock-On Time						25	ms
PULSE DENSITY MODULATION (PDM) OUTPUTS							
Dynamic Range[1]		A-weighted filter			126		dB
THD+N Level[3]		-6dBFS input level			-125		dBFS

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
Group Delay[5]		f_S = 384kHz, f_{PDM_CLK} = 6.144MHz			7.5		µs
		f_S = 768kHz, f_{PDM_CLK} = 6.144MHz			4.9		
MASTER CLOCK TIMING SPECIFICATIONS							
Master Clock Period	t_{MP}	900kHz to 49.152MHz input clock frequency		0.0203		1.11	µs
SERIAL AUDIO DATA PORT TIMING SPECIFICATIONS							
Nominal Bit Clock Frequency	f_{BCLK}	Maximum valid frequency	Slave mode		49.152		MHz
Nominal Bit Clock Frequency Range	f_{BCLK}	Maximum valid frequency	Master mode		24.576		MHz
		Minimum valid frequency	Slave mode		0.512		
			Master mode		2.048		
Bit Clock Low Pulse Width	t_{BL}	Master or slave mode		18			ns
Bit Clock High Pulse Width	t_{BH}	Master or slave mode		18			ns
Frame Sync Clock Frequency Range	f_{SYNC}	Maximum setting			768		kHz
		Minimum setting			8		
Frame Sync to Bit Clock Active Edge Setup Time	t_{LS}	Slave mode, setup time to bit clock active edge		3			ns
Bit Clock Active Edge to Frame Sync Hold Time	t_{LH}	Slave mode, hold time from bit clock active edge		5			ns
Data Input to Bit Clock Active Edge Setup Time	t_{SS}	Master or slave mode, setup time to bit clock active edge		3			ns
Bit Clock Active Edge to Data Input Hold Time	t_{SH}	Master or slave mode, hold time from bit clock active edge		10			ns
Bit Clock Inactive Edge to Frame Sync Edge Timing Skew	t_{TS}	Master mode				6	ns
Bit Clock Inactive Edge to Data Output Delay	t_{SOD}	Master or slave mode, delay until	V_{IOVDD} ≥ 1.62V	0		16	ns

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
		data output logic level change					
		Master or slave mode, delay until data output level change	V_{IOVDD} ≥ 1.1V	0		32	
Bit Clock Inactive Edge to Data Output Active	t_{SOTD}	Master or slave mode, delay until high-Z data output is active		0		16	ns
Bit Clock Inactive Edge to Data Output High-Z	t_{SOTX}	Master or slave mode, delay until active data output is high-Z		0		16	ns
I²C CONTROL PORT TIMING SPECIFICATIONS							
Serial Clock (SCL) Frequency	f_{SCL}					1	MHz
SCL Pulse Width High	t_{SCLH}			0.26			µs
SCL Pulse Width Low	t_{SCLL}			0.5			µs
SCL Setup Time for a Repeated Start Condition	t_{SCS}	Setup time from SCL rising to SDA falling		0.26			µs
SCL Hold Time for a Start Condition	t_{SCH}	Hold time from SDA falling to SCL falling		0.26			µs
SCL Setup Time for a Stop Condition	t_{BFT}	Setup time from SCL rising to SDA rising		0.5			µs
Serial Data (SDA) Setup Time	t_{DS}	SDA setup time to SCL rising		50			ns
Serial Data (SDA) Hold Time	t_{DH}	SDA hold time from SCL falling		0			ns
SCL and SDA Rise Time	t_{SCR}	400pF load				120	ns
SCL and SDA Fall Time	t_{SCF}	400pF load				120	ns
Bus Free Time Between Stop and Start Condition	t_{BUF}	Master mode		0.5			µs
I³C CONTROL PORT TIMING SPECIFICATIONS							
Maximum Serial Clock (SCL) Frequency	f_{SCL}	Master or slave mode			6.144		MHz

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS		MIN	TYP	MAX	UNITS
Maximum Serial Data (SDA) Rate	f_{SDA}	Master DDR mode			6.144		MHz
SCL Pulse Width High	t_{SCLH}	Slave mode		24			ns
SCL Pulse Width Low	t_{SCLL}	Slave mode		24			ns
SDA Input Setup Time	t_{DS}	Setup time from SDA edge to SCL rising edge	Slave mode	19			ns
			Master mode	10			
SDA Input Hold Time	t_{DH}	Slave mode		11			ns
SERIAL PERIPHERAL INTERFACE (SPI) PORT TIMING SPECIFICATIONS							
Serial Clock Frequency	f_{SCLK}					10	MHz
Serial Clock Pulse Width	t_{CCPL}	Low pulse		35			ns
	t_{CCPH}	High pulse		35			
Slave Select to Serial Clock Setup Time	t_{CLS}	Setup time to serial clock rising edge		5			ns
Serial Clock to Slave Select Hold Time	t_{CLH}	Hold time from serial clock rising		40			ns
Slave Select Pulse Width	t_{CLPH}	Minimum high pulse		10			ns
MOSI to Serial Clock Setup Time	t_{CDS}	Setup time to serial clock rising		10			ns
Serial Clock to MOSI Hold Time	t_{CDH}	Hold time from serial clock rising		10			ns
Serial Clock to MISO Data Delay Time	t_{COD}	Delay until MISO level change				30	ns
Slave Select to MISO High-Z Delay Time	t_{COTS}	Delay until MISO becomes high-Z				30	ns
QUAD-SPI (QSPI) MASTER PORT TIMING SPECIFICATIONS							
QSPI Clock Frequency	f_{SCLK}	QSPI output clock (QSPI_CLK) during self-boot				12.5	MHz
QSPI Data Input Setup Time	t_{CDS}	QSPI data input to clock rising edge setup time		10			ns
QSPI Data Input Hold Time	t_{CDH}	QSPI clock rising to data input edge hold time		10			ns

(Supply voltages V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V and V_{DVDD} = 0.9V (external), Supply bypass C_{AVDD} = C_{HPVDD} = C_{IOVDD} = 2.2µF + 2 x 0.1µF and C_{DVDD} = 2.2µF + 3 x 0.1µF, Master clock = 24.576MHz, Serial audio data port sample rate = 48kHz, Audio data word width = 24-bits, Headphone amplifier load = 16Ω +33µH, default analog power modes, AC measurement bandwidth = 20Hz to 20kHz, Ambient temperature (T_A) = -40°C to +85°C with typical values at +25°C unless otherwise noted, see the *System Block Diagram* for other external component values.)

PARAMETER	SYMBOL	CONDITIONS/COMMENTS	MIN	TYP	MAX	UNITS
QSPI Data Output Delay Time	t_{COD}	QSPI data output delay from clock falling edge	1			ns
QSPI Data Output High-Z Delay Time	t_{CODS}	QSPI data output high-Z delay from chip select rising edge	1			ns
UNIVERSAL ASYNCHRONOUS RECEIVER-TRANSMITTER (UART) PORT TIMING SPECIFICATIONS						
UART Baud Rate					3.125	Mbps
ACTIVE-LOW HARDWARE POWER-DOWN INPUT PIN TIMING SPECIFICATIONS						
Power-Down Input Assert Time	t_{RLPW}	Minimum time $\overline{PD}$ input must be asserted low to power down the device	20			ns
Power-Down Input Hardware Enable Time		REG_EN is pulled high, f_{MCLK} = 24.576MHz	15			ms
GENERAL-PURPOSE INPUT/OUTPUT (GPIO) PIN TIMING SPECIFICATIONS						
GPIO (MPx) Input Latency	t_{GIL}	Time delay until MPx logic level is read internally			1.5 x $1/f_S$	µs
DIGITAL MICROPHONE INTERFACE TIMING SPECIFICATIONS						
DMIC Clock Output Fall Time	t_{CF}	2mA output drive strength, 25pF load			12	ns
DMIC Clock Output Rise Time	t_{CR}	2mA output drive strength, 25pF load			14	ns
DMIC Data Setup time	t_{SETUP}	Setup time from DMIC data edge to DMIC clock edge	10			ns
DMIC Data Hold Time	t_{HOLD}	Hold time from PDM clock edge to DMIC data edge	3			ns
PULSE DENSITY MODULATION (PDM) OUTPUT TIMING SPECIFICATIONS						
PDM Clock Frequency	f_{PDM_CLK}	PDM_RATE = 0		12.288		MHz
		PDM_RATE = 1		6.144		
		PDM_RATE = 2		3.072		
PDM Clock Output Fall Time	t_{CF}	2mA output drive strength, 25pF load			12	ns
PDM Clock Output Rise Time	t_{CR}	2mA output drive strength, 25pF load			14	ns
PDM Data Hold Time	t_{HOLD}	Delay time from PDM clock edge to PDM data change	35		46	ns

[1] Dynamic range is the ratio of the sum of the inband noise and harmonic power with a -60dBFS input signal level at 1kHz relative to the full-scale power level in decibels. Normal operating mode uses default settings, while extreme power saving mode changes ADCx_IBIAS to 0x1 and ADC_LP_MODE to 1.

[2] SNR is the ratio of the sum of all inband noise power with no input signal relative to the full-scale power level in decibels.

[3] THD+N level is the ratio of the sum of the inband harmonic power with the specified input signal level at 1kHz relative to either full-scale code (in dBFS) or a $1V_{RMS}$ reference level (in dBV).

[4] Group delay specified from analog input to Class-D amplifier output with FastDSP in bypass mode, ADCxx_FCOMP = 0, and DAC_FCOMP = 0. Point group delay values taken at f_{IN} = 1kHz, see the group delay Typical Performance Characteristics for the delay over the entire audio band.

[5] Group delay specified from input to the PDM output channels to the PDM output pins. Point group delay values taken at f_{IN} = 1kHz at the specified PDM sample rate.

[6] ASRC THD+N typical value is specified at f_{IN} = 1kHz and the maximum value is specified at f_{IN} = 20kHz

Timing Diagrams

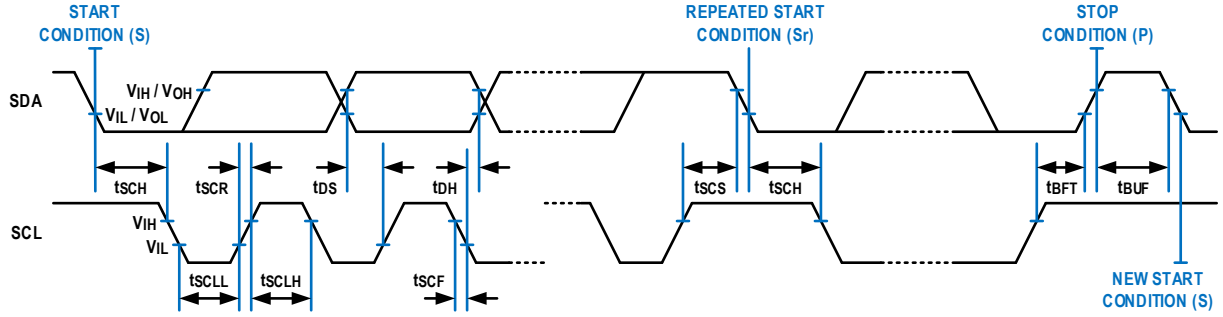

Figure 2. I²C Interface Slave Mode Timing Diagram

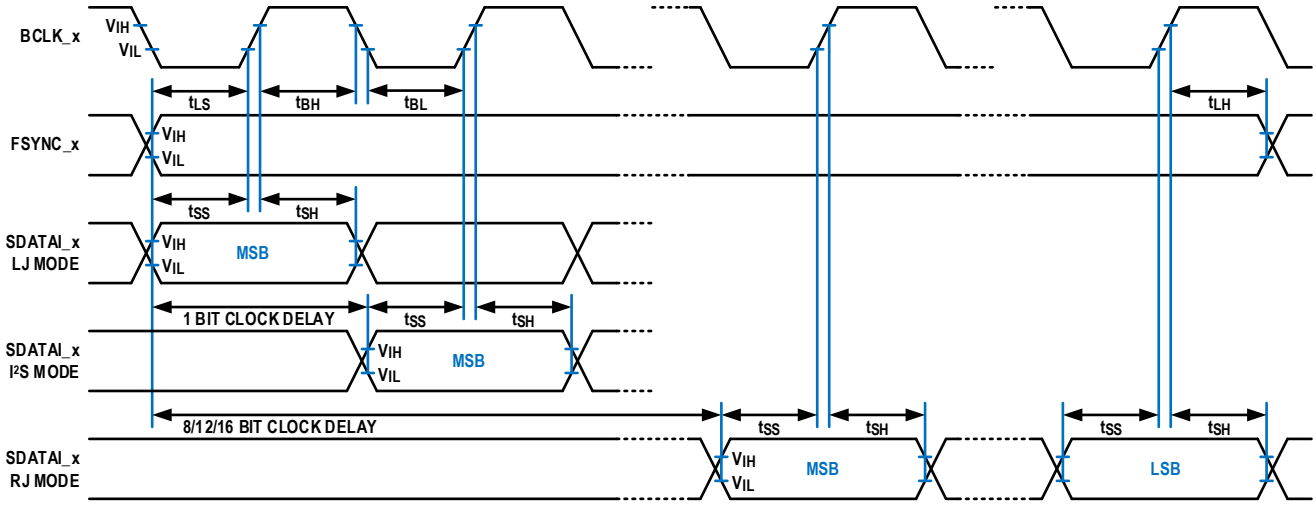

Figure 3. Serial Audio Data Port Input Timing (Slave Mode, Bit Clock Rising Active Edge)

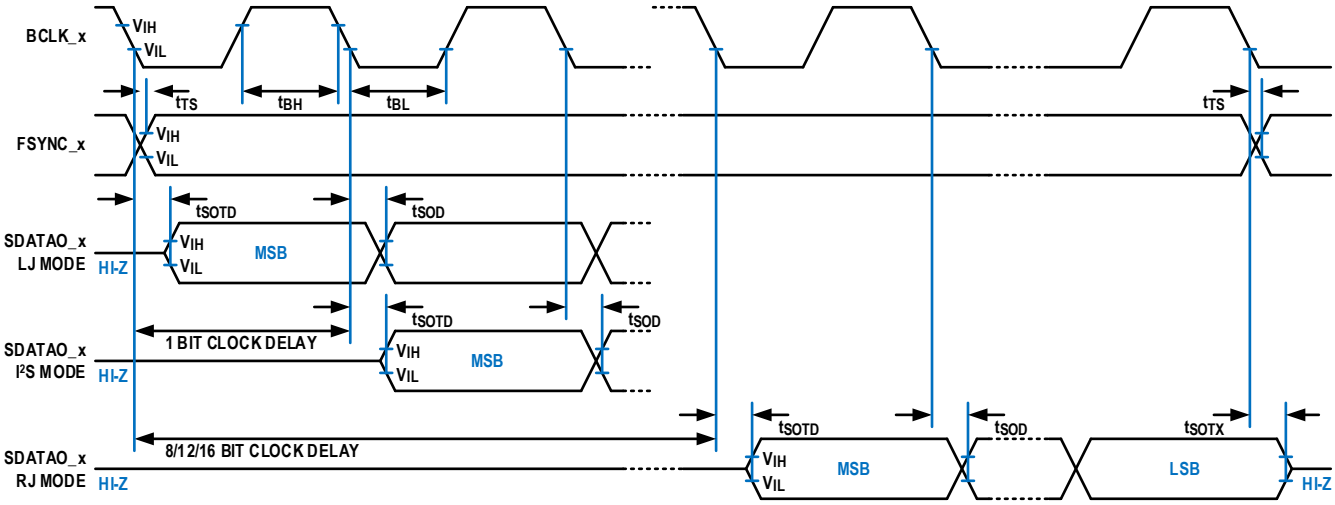

Figure 4. Serial Audio Data Port Output Timing (Master Mode, Bit Clock Rising Active Edge, Tristate Enabled)

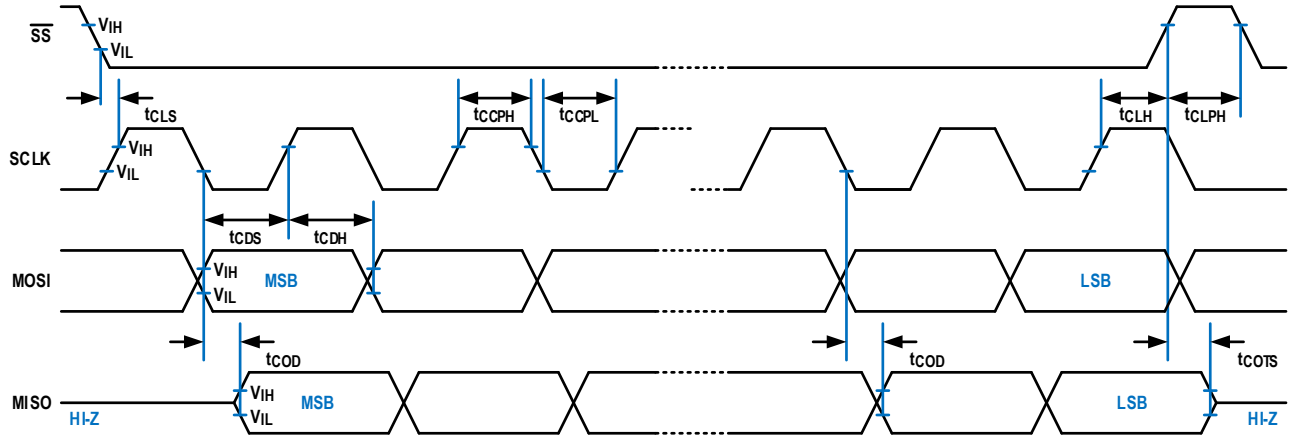

Figure 5. SPI Port Timing Diagram

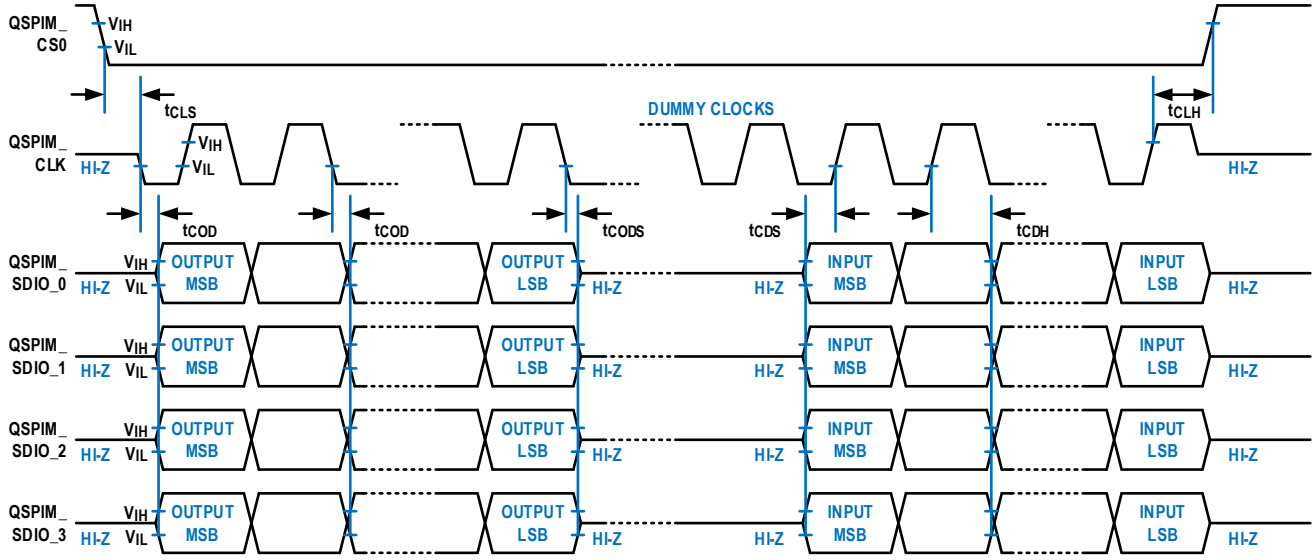

Figure 6. QSPI Master Port Timing Diagram (Default Settings with Tristate Idle)

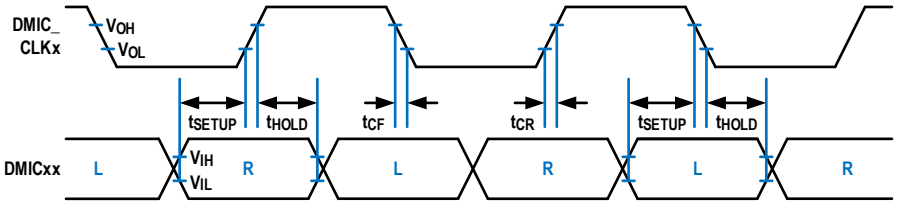

Figure 7. Digital Microphone Timing Diagram

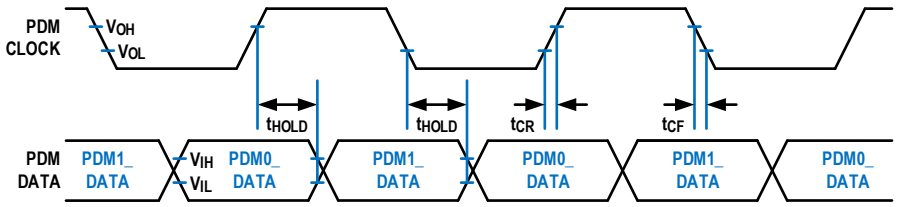

Figure 8. PDM Output Timing Diagram

Power Consumption Specifications

Device Power-Down Modes

Supplies externally connected at $V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V$ and $V_{DVDD} = 0.9V$, PLL and crystal oscillator disabled.

Table 2. Quiescent Power Consumption for Power-Down Mode Configurations

DEVICE STATE	CONDITIONS	TYPICAL AVDD CURRENT	TYPICAL HPVDD CURRENT	TYPICAL DVDD CURRENT	TYPICAL IOVDD CURRENT	UNIT
Hardware Power-Down	$\overline{PD}$ pin low	0.52	0.8	30.2	0.02	µA
Software Power-Down	$\overline{PD}$ pin high, POWER_EN = 0, no keep-alive modes enabled	1.08	1.35	31.17	0.02	µA
Software Power-Down with Common-Mode Keep Alive Set	$\overline{PD}$ pin high, POWER_EN = 0, CM_KEEP_ALIVE = 1, KEEP_MEM = 0	63.92	1.35	32.76	0.02	µA
Software Power-Down with Common-Mode, FastDSP/HiFi 3z Memory Keep Alive Set	$\overline{PD}$ pin high, POWER_EN = 0, CM_KEEP_ALIVE = 1, KEEP_MEM = 1	69.93	1.35	32.78	0.02	µA

Power Management Mode Configurations

Configuration models a typical active noise canceling (ANC) use case with different analog power mode settings. External supplies are $V_{AVDD} = V_{HPVDD} = V_{IOVDD} = 1.8V$ and $V_{DVDD} = 0.9V$. PLL enabled with $f_{MCLK} = 24.576MHz$ (crystal amp enabled). 3 ADC inputs with PGAs in single-ended mode with channels 0 and 1 set to $f_S = 384kHz$ and channel 2 set to $f_S = 48kHz$. 2 interpolator and decimator channels. FastDSP core at $f_S = 384kHz$, FDSP_SPEED = 0 (24.576MHz), and 64 instructions. HiFi 3z core disabled. All biquad filters are set to 27-bit precision. DAC to Class-D amp output channel set to $f_S = 384kHz$ with a 16Ω load. Single serial audio data port in slave mode for input and output with 1 input and 2 output ASRC channels. Quiescent current state (zero code or no signal input to all audio channels).

Table 3. Quiescent Power Consumption Comparison for Power Management Modes

POWER MODE	POWER MANAGEMENT DEVICE SETTINGS	I_{DVDD} (mA)	I_{AVDD} (mA)	I_{HPVDD} (mA)	I_{IOVDD} (mA)	TOTAL POWER (mW)	ADC A-wt DR (dB)	ADC THD+N (dB)[1]	Class-D A-wt DR (dB)	Class-D THD+N (dB)[1]
Normal Power Mode	Default Settings	10.141	1.307	0.747	0.693	14.07	100.0	-90.0	110.0	-90.9
Power-Savings Mode	ADC_IBIAS: 0x010, ADC_LP_MODE: 1, DAC_IBIAS: 0x01, DAC_PWR_MODE: 0x01, ASRCI_LPM1: 1, ASRCI_LPM2: 0, PGA_IBIAS: 0x10, ASRCO_LPM1: 1, ASRCO_LPM2: 0,	9.931	1.172	0.735	0.709	13.65	98.0	-86.3	112.4	-90.6
Extreme Power-Savings Mode	ADC_IBIAS: 0x001, ADC_LP_MODE: 1, DAC_IBIAS: 0x01, DAC_PWR_MODE: 0x10, ASRCI_LPM1: 0, ASRCI_LPM2: 1, PGA_IBIAS: 1, ASRCO_LPM1: 0, ASRCO_LPM2: 1	9.869	1.083	0.675	0.71	13.324	97.9	-85.6	112.4	-90.8

[1] THD+N ratio is measured at $f_{IN} = 1kHz$ with a -1dBFS output for ADC input channels and at $f_{IN} = 1kHz$ with 50mW into 16Ω for Class-D amplifier.

ABSOLUTE MAXIMUM RATINGS

T_A = +25°C unless otherwise specified.

Table 4. Absolute Maximum Ratings

PARAMETER	RATING
Power Supply (AVDD, HPVDD IOVDD)	-0.3V to +1.98V
Digital Supply (DVDD)	-0.3V to +1.98V
Input Current (Except Supply Pins)	±20mA
Analog Input Voltage (Signal Pins)	-0.3V to V_{AVDD} + 0.3V
Digital Input Voltage (Signal Pins)	-0.3V to V_{IOVDD} + 0.3V
Operating Temperature Range (Case)	-40°C to +85°C
Storage Temperature Range	-65°C to +150°C

Stresses beyond those listed under Absolute Maximum Ratings may cause permanent damage to the device. These are stress ratings only, functional operation of the product at these or any other conditions above those indicated in the operational section of this specification is not implied. Operation beyond the maximum operating conditions for extended periods may affect product reliability.

Thermal Resistance

Thermal performance is directly linked to PCB design and operating environment. Close attention to PCB thermal design is required. θ_{JA} and θ_{JC} are determined using JESD51-9 on a 4-layer PCB with natural convection cooling.

Table 5. Thermal Resistance

PACKAGE TYPE	θ_{JA}[1]	θ_{JC}[1]	UNIT
CB-77-1	17	0.3	°C/W

[1] Thermal impedance simulated values are based on a JEDEC 2S2P thermal test board with two thermal vias. See JEDEC JESD-51.

Electrostatic Discharge (ESD)

The following ESD information is provided for the handling of ESD-sensitive devices in an ESD-protected area-only Human body model (HBM) per ANSI/ESDA/JEDEC JS-001 Field-induced charged device model (FICDM) and charged device model (CDM) per ANSI/ESDA/JEDEC JS-002. International Electrotechnical Commission (IEC) electromagnetic compatibility: Part 4-2 (IEC) per IEC 61000-4-2. Machine model (MM) per ANSI/ESD STM5.2. MM voltage values are for characterization only.

Table 6. ESD Ratings

ESD MODEL	WITHSTAND THRESHOLD (V)	CLASS
HBM	±7000	3A
CDM	±1500	C3
IEC[1]	±12000 (contact discharge)	Level 4
	>±12000 (air discharge)	Level 3
MM	±200	M3

[1] The IEC withstand threshold applies to the A, B, Y, and Z pins. The IEC withstand thresholds are for 10 positive and 10 negative discharges. The withstand voltage for three positive and three negative discharges is ±15,000 V; IEC Level 4.

ESD Caution

ESD (electrostatic discharge) sensitive device. Charged devices and circuit boards can discharge without detection. Although this product features patented or proprietary protection circuitry, damage may occur on devices subjected to high-energy ESD. Therefore, proper ESD precautions should be taken to avoid performance degradation or loss of functionality.

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

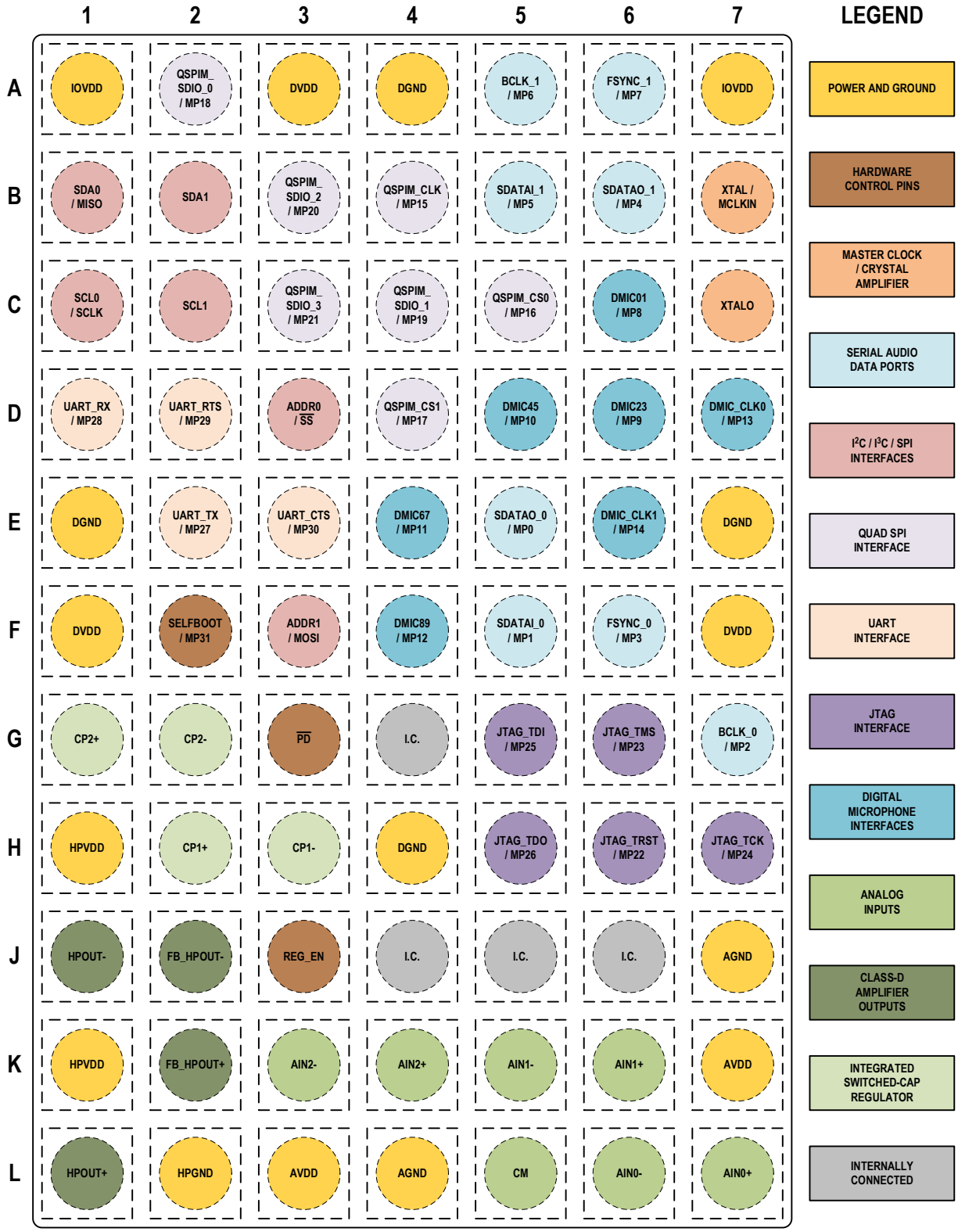

Figure 9. Top View Ball Configuration (View from Top-Side)

Table 7. Pin Function Descriptions

PIN	NAME	DESCRIPTION	Type
A1	IOVDD	Digital Input and Output Pin Supply. The digital output pins are supplied from IOVDD, and this pin sets the highest input voltage seen on the digital input pins. The current draw of this pin is variable because the current is dependent on the loads of the digital outputs. Decouple each IOVDD input to DGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
A2	QSPIM_ SDIO_0 / MP18	Quad Master SPI Data I/O (QSPIM_SDIO_0) or Multipurpose I/O 18	D_IO
A3	DVDD	Digital Core Supply. The digital supply can be generated from an on-chip regulator or supplied directly from an external supply. Decouple each DVDD input to DGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
A4	DGND	Digital Ground. The AGND and DGND pins can be tied directly together in a common ground plane.	PWR
A5	BCLK_1 / MP6	Serial Audio Data Port 1 Bit Clock or Multipurpose I/O 6	D_IO
A6	FSYNC_1 / MP7	Serial Audio Port 1 Frame Sync/Left Right Clock (FSYNC_1) or Multipurpose I/O 7	D_IO
A7	IOVDD	Digital Input and Output Pin Supply. The digital output pins are supplied from IOVDD, and this pin sets the highest input voltage seen on the digital input pins. The current draw of this pin is variable because the current is dependent on the loads of the digital outputs. Decouple each IOVDD input to DGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
B1	SDA0/MISO	I^2C Data 0 I/O (SDA0) or SPI Data Output (MISO). In I^2C mode, this pin is a bidirectional open-drain input. The line connected to this pin must have a 2.0kΩ pullup resistor. In SPI mode, the SPI data output is used for reading back registers and memory locations. This pin is tri-stated when an SPI read is not active.	D_IO
B2	SDA1	I^2C/I^3C Master Data 1 I/O	D_IO
B3	QSPIM_ SDIO2 / MP20	Quad Master SPI Data I/O 2 (QSPIM_SDIO2) or Multipurpose I/O 20	D_IO
B4	QSPIM_ CLK / MP15	Quad Master SPI Clock (QSPIM_CLK) or Multipurpose I/O 15	D_IO
B5	SDATAI_1 / MP5	Serial Audio Port 1 Input Data (SDATAI_1) or Multipurpose I/O 5	D_IO
B6	SDATAO_1 / MP4	Serial Audio Port 1 Output Data (SDATAO_1) or Multipurpose I/O 4	D_IO
B7	XTALI / MCLKIN	Crystal Clock Input (XTALI)/Master Clock Input (MCLKIN)	D_IN

C1	SCL0 / SCLK	I^2C Clock 0 (SCL0) or SPI Clock (SCLK). In I^2C mode, this pin is an open-collector input. When the device is in self-boot mode, this pin is an open-collector output (I^2C master). In I^2C mode, the line connected to this pin must have a 2.0kΩ pullup resistor. In SPI mode, this pin is the SPI Clock. This pin can either run continuously or be gated off between SPI transactions.	D_IN
C2	SCL1	I^2C/I^3C Master Clock Output	D_O
C3	QSPIM_ SDI03 / MP21	Quad Master SPI Data I/O 3 (QSPIM_SDIO3) or Multipurpose I/O 21	D_IO
C4	QSPIM_ SDIO1 / MP19	Quad Master SPI Data I/O 1 (QSPIM_SDIO1) or Multipurpose I/O 19	D_IO
C5	QSPIM_CS0 / MP16	Quad Master SPI Chip Select 0 (QSPIM_CS0) or Multipurpose I/O 16	D_IO
C6	DMIC01 / MP8	Digital Microphone Stereo Input 0 and 1 (DMIC01) or Multipurpose I/O 8	D_IO
C7	XTALO	Crystal Clock Amp Output. This pin is the output of the crystal amplifier. Do not use this pin to provide a clock to other ICs in the system.	A_OUT
D1	UART_RX / MP28	UART Port Data Receiver Input (UART_RX) or Multipurpose I/O 28	D_IO
D2	UART_RTS / MP29	UART Port Flow Control Ready to Send Output (UART_RTS) or Multipurpose I/O 29	D_IO
D3	ADDR0 / $\overline{SS}$	I^2C Address 0 (ADDR0) or SPI Latch Signal ($\overline{SS}$). In I^2C mode, ADDR0 and ADDR1 are used to select one of four I^2C address options. In SPI mode, this pin must go low at the beginning of a transaction and high at the end of a transaction. Each SPI transaction may take a different number of SCLK cycles to complete, depending on the address and read/write bit that is sent at the beginning of the SPI transaction.	D_IN
D4	QSPIM_CS1 / MP17	Quad Master SPI Chip Select 1 (QSPIM_CS1) or Multipurpose I/O 17	D_IO
D5	DMIC45 / MP10	Digital Microphone Stereo Input 4 and 5 (DMIC45) or Multipurpose I/O 10	D_IO
D6	DMIC23 / MP9	Digital Microphone Stereo Input 2 and 3 (DMIC23) or Multipurpose I/O 9	D_IO
D7	DMIC_CLK0 / MP13	Digital Microphone Clock Output 0 (DMIC_CLK0) or Multipurpose I/O 13	D_IO
E1	DGND	Digital Ground. The AGND and DGND pins can be tied directly together in a common ground plane.	PWR
E2	UART_TX / MP27	UART Port Data Transmit Output (UART_TX) or Multipurpose I/O 27	D_IO

E3	UART_CTS / MP30	UART Port Flow Control Clear to Send Input (UART_CTS) or Multipurpose I/O 30	D_IO
E4	DMIC67 / MP11	Digital Microphone Stereo Input 6 and 7 (DMIC67) or Multipurpose I/O 11	D_IO
E5	SDATAO_0 / MP0	Serial Audio Port 0 Output Data (SDATAO_0) or Multipurpose I/O 0	D_IO
E6	DMIC_CLK1 / MP14	Digital Microphone Clock Output 1 (DMIC_CLK1) or Multipurpose I/O 14	D_IO
E7	DGND	Digital Ground. The AGND and DGND pins can be tied directly together in a common ground plane.	PWR
F1	DVDD	Digital Core Supply. The digital supply can be generated from an on-chip regulator or supplied directly from an external supply. Decouple each DVDD input to DGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
F2	SELFBOOT / MP31	Self-Boot Select or Multipurpose I/O 31. Connect this pin to IOVDD through a 100kΩ resistor at power-up to enable the self-boot mode. This pin also outputs the buffered Crystal Oscillator clock by default at the start-up. Otherwise, set this pin to DGND through a 100kΩ resistor. Once the power-up is completed, this pin can be re-configured as Multipurpose I/O 31.	D_IN
F3	ADDR1 / MOSI	I^2C Address 1 (ADDR1) or SPI Data Input (MOSI). In I^2C mode, ADDR0 and ADDR1 are used to select one of four I^2C address options. In SPI mode, the SPI data input is used for writing registers and memory locations.	D_IN
F4	DMIC89 / MP12	Digital Microphone Stereo Input 8 and 9 (DMIC89) or Multipurpose I/O 12	D_IO
F5	SDATAI_0 / MP1	Serial Audio Port 0 Input Data (SDATAI_0)/Multipurpose I/O 1	D_IO
F6	FSYNC_0 / MP3	Serial Audio Port 0 Frame Sync/Left Right Clock (FSYNC_0) or Multipurpose I/O 3	D_IO
F7	DVDD	Digital Core Supply. The digital supply can be generated from an on-chip regulator or supplied directly from an external supply. Decouple each DVDD input to DGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
G1	CP2+	Switched-Cap Regulator Capacitor 2 Positive Connection. Connect a 1µF capacitor between CP2+ and CP2-.	PWR
G2	CP2-	Switched-Cap Regulator Capacitor 2 Negative Connection. Connect a 1µF capacitor between CP2+ and CP2-.	PWR
G3	$\overline{\text{PD}}$	Active Low Power-Down Input. All digital and analog circuits are powered down. The external pulldown resistor to DGND is recommended on this pin to hold the device in power-down mode if the input signal from the system micro-controller is floating while power is applied to the supply pins.	D_IN
G4	I.C.	Internally Connected. Connect to DGND.	DGND

G5	JTAG_TDI / MP25	JTAG Debug Port Data Input (JTAG_TDI) or Multipurpose I/O 25	D_IO
G6	JTAG_TMS / MP23	JTAG Debug Port Master Select (JTAG_TMS) or Multipurpose I/O 23	D_IO
G7	BCLK_0 / MP2	Serial Audio Port 0 Bit Clock (BCLK_0) or Multipurpose I/O 2	D_IO
H1	HPVDD	Headphone Amplifier 1.8V Analog Power Supply. The PCB trace to this pin must be wider to supply the higher current necessary for driving the headphone outputs. Decouple each HPVDD input to HPGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor to provide the peak current necessary for low frequency signals.	PWR
H2	CP1+	Switched-Cap Regulator Capacitor 1 Positive Connection. Connect a 1µF capacitor between CP1+ and CP1-.	PWR
H3	CP1-	Switched-Cap Regulator Capacitor 1 Negative Connection. Connect a 1µF capacitor between CP1+ and CP1-.	PWR
H4	DGND	Digital Ground. The AGND and DGND pins can be tied directly together in a common ground plane.	PWR
H5	JTAG_TDO / MP26	JTAG Debug Port Data Output (JTAG_TDO) or Multipurpose I/O 26	D_IO
H6	JTAG_TRST / MP22	JTAG Debug Port Reset (JTAG_TRST) or Multipurpose I/O 22	D_IO
H7	JTAG_TCK / MP24	JTAG Debug Port Clock (JTAG_TCK) or Multipurpose I/O 24	D_IO
J1	HPOUT-	Headphone Output Inverted	A_OUT
J2	FB_HPOUT-	Headphone Output Inverted Feedback Signal. Connect close to the inverted side of the headphone load after any filtering components.	A_IN
J3	REG_EN	Regulator Enable. Tie this pin to HPVDD to enable the internal DVDD regulator and tie this pin to ground to disable the regulator.	A_IN
J4	I.C.	Internally Connected. Leave this pin unconnected.	DNC
J5	I.C.	Internally Connected. Connect to DGND.	DGND
J6		Internally Connected. Leave this pin unconnected.	DNC
J7	AGND	Analog Ground. The AGND and DGND pins can be tied directly together in a common ground plane.	PWR
K1	HPVDD	Headphone Amplifier 1.8V Analog Power Supply. The PCB trace to this pin must be wider to supply the higher current necessary for driving the headphone outputs. Decouple each HPVDD input to HPGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor to provide the peak current necessary for low frequency signals.	PWR
K2	FB_HPOUT+	Headphone Output Noninverted Feedback Signal. Connect close to the noninverted side of the headphone load after any filtering components.	A_IN

K3	AGND	Analog Ground. The AGND and DGND pins can be tied directly together in a common ground plane.	PWR
K4	AIN2+	Analog Input ADC2 Non-Inverting Input	A_IN
K5	AIN1-	Analog Input ADC1 Inverting Input	A_IN
K6	AIN1+	Analog Input ADC1 Non-Inverting Input	A_IN
K7	AVDD	1.8V Analog Supply. Decouple each AVDD input to AGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
L1	HPOUT+	Headphone Output Noninverted	A_OUT
L2	HPGND	Headphone Amplifier and Regulator Ground	PWR
L3	AVDD	1.8V Analog Supply. Decouple each AVDD input to AGND with a 0.1µF capacitor, and a single shared 2.2µF bulk capacitor.	PWR
L4	AIN2-	Analog Input ADC2 Inverting Input	A_IN
L5	CM	Common-Mode Reference Output. The CM output is fixed at 0.85V nominal. Connect a 10µF and 0.1µF decoupling capacitor between this pin and AGND to reduce crosstalk between the ADC channels. The material of the capacitors is not critical. This pin can supply a reference bias to external analog circuits as long as they are not drawing current from the CM output (Example: high impedance input of an external amplifier).	A_OUT
L6	AIN0-	Analog Input ADC0 Inverting Input	A_IN
L7	AIN0+	Analog Input ADC0 Non-Inverting Input	A_IN

TYPICAL PERFORMANCE CHARACTERISTICS

See the *Specifications* table for the overall global conditions.

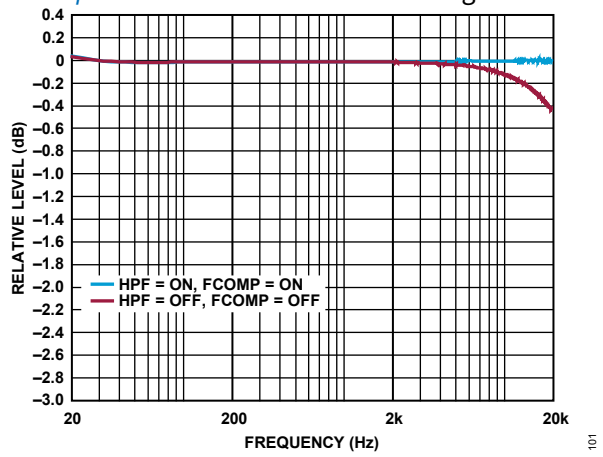

Figure 10. Frequency Response, Fs = 48kHz, -20dBV Input, Signal Path = AINx to SDATAO_x, No PGA

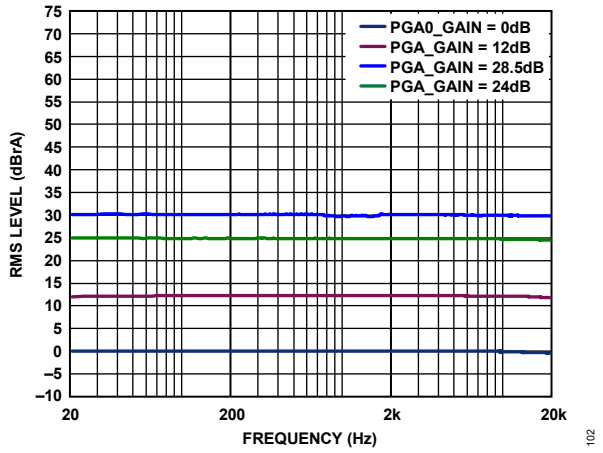

Figure 11. Frequency Response, Fs = 48kHz, Signal Path = AINx to SDATAO_x

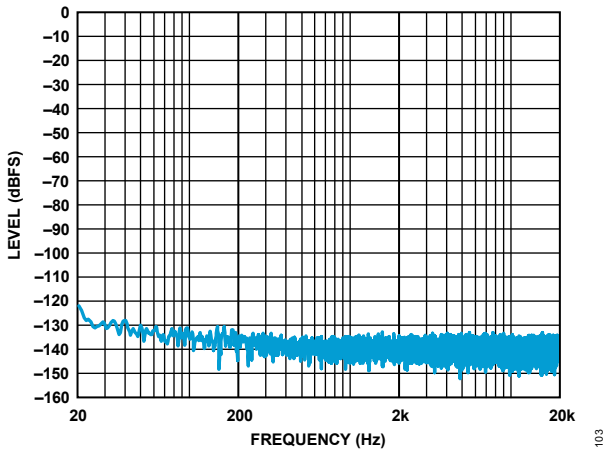

Figure 12. FFT, No Signal, Fs = 48kHz, Signal Path = AINx to SDATAO_x, No PGA

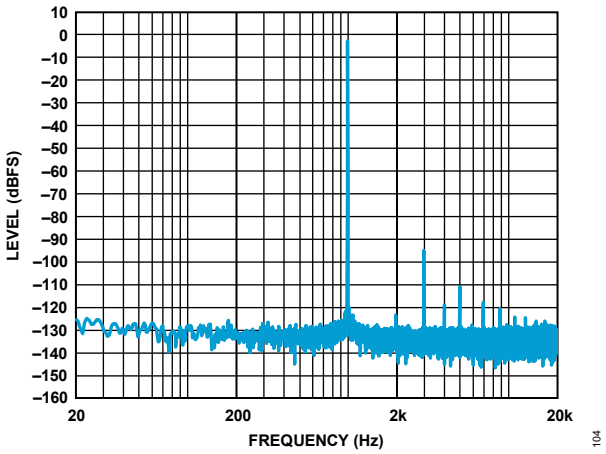

Figure 13. FFT, -7dBV Input, Fs = 48kHz, Signal Path = AINx to SDATAO_x, No PGA

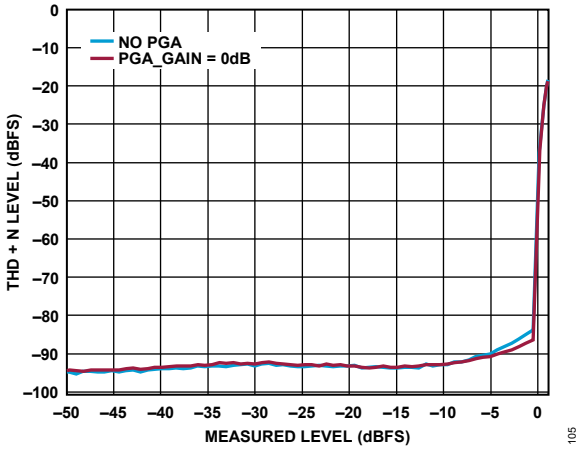

Figure 14. THD + N Level vs. Amplitude, Fs = 48kHz, Signal Path = AINx to SDATAO_x

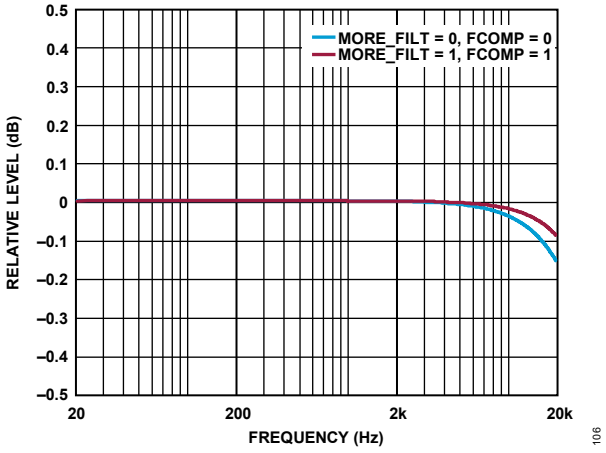

Figure 15. Frequency Response, Fs = 48kHz, Signal Path = SDATAI_x to PDMOUT

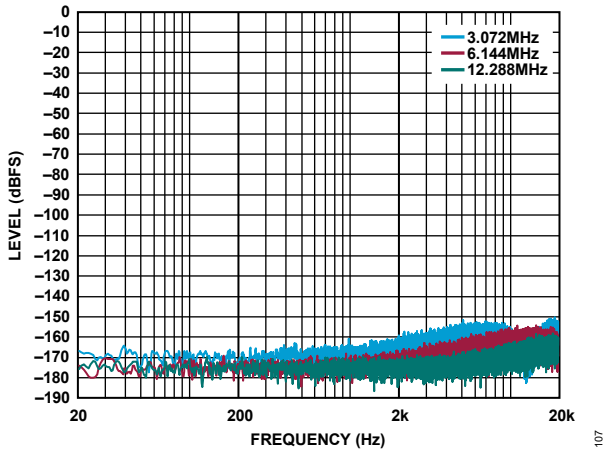

Figure 16. FFT, No Signal, Fs = 48kHz, Signal Path = SDATAI_x to FDSP to PDMOUT

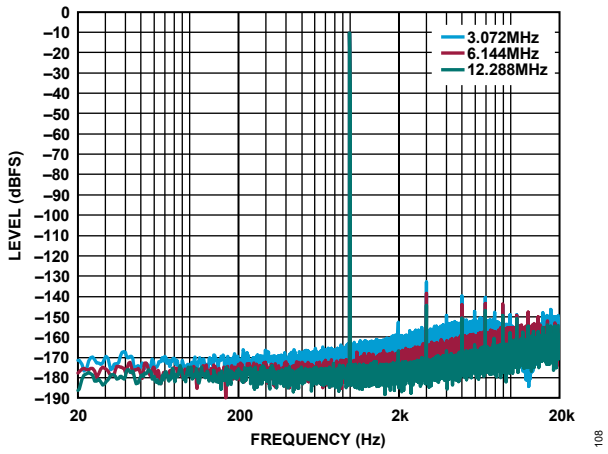

Figure 17. FFT, -10dBFS input, Fs = 48kHz, Signal Path = SDATAI_x to FDSP to PDMOUT

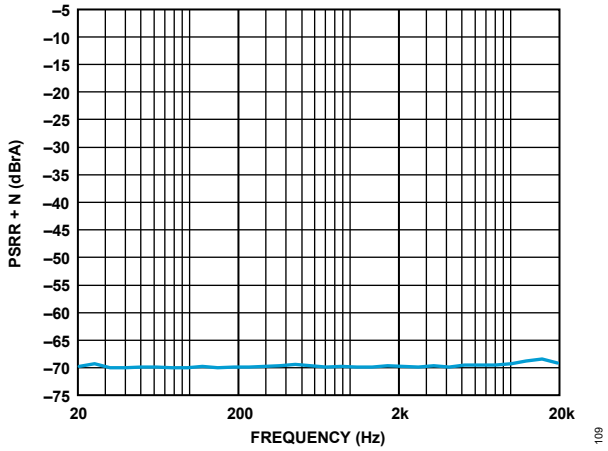

Figure 18. PSRR + N, Signal Path = AINx to SDATAO_x, Fs = 48kHz, 100mVp-p Ripple on AVDD, No PGA

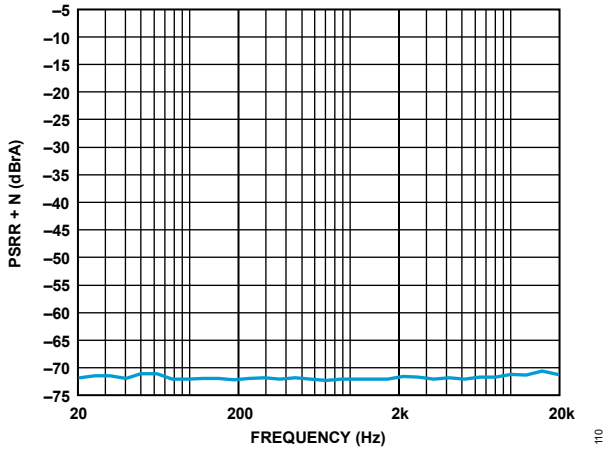

Figure 19. PSRR + N, Signal Path = AINx to SDATAO_x, Fs = 48kHz, 100mVp-p Ripple on AVDD, PGA = 0dB Gain

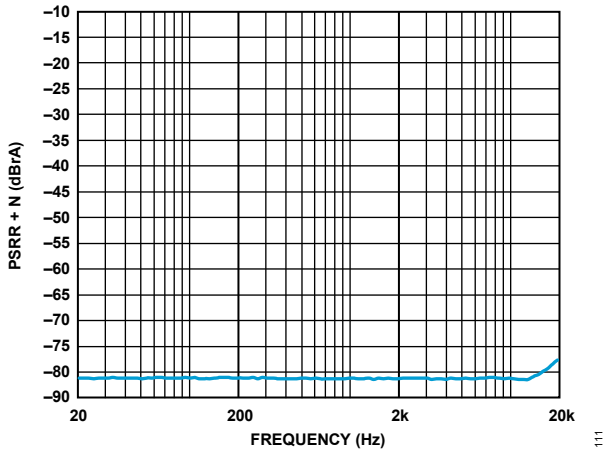

Figure 20. PSRR + N, Signal Path = SDATAI_x to HPOUT, Fs = 48kHz, 100mVp-p Ripple on HPVDD

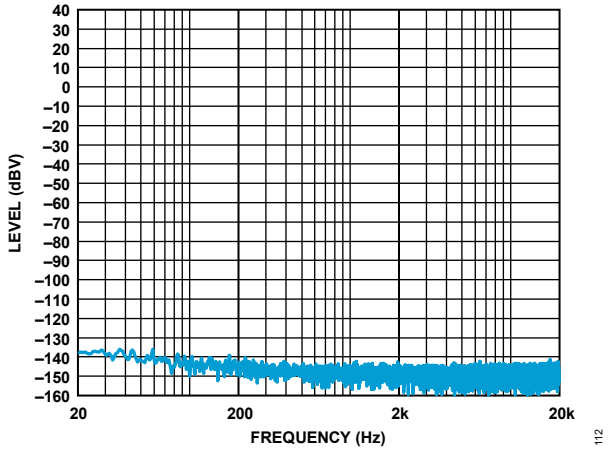

Figure 21. FFT, No Signal, Fs = 48kHz, Signal Path = SDATAI_x to HPOUT, 16Ω Load

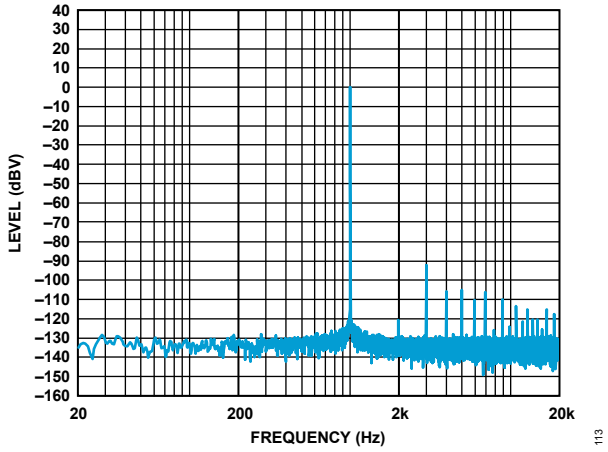

Figure 22. FFT, -1dBFS Input, Fs = 48kHz, Signal Path = SDATAI_x to HPOUT, 32Ω Load

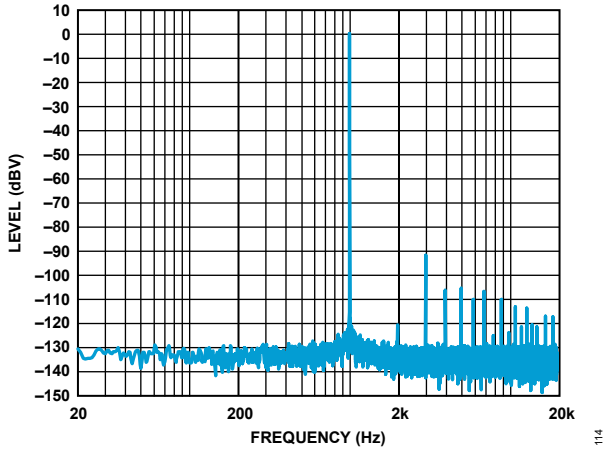

Figure 23.FFT, -1dBFS Input, Fs = 48kHz, Signal Path = SDATAI_x to HPOUT, 24Ω Load

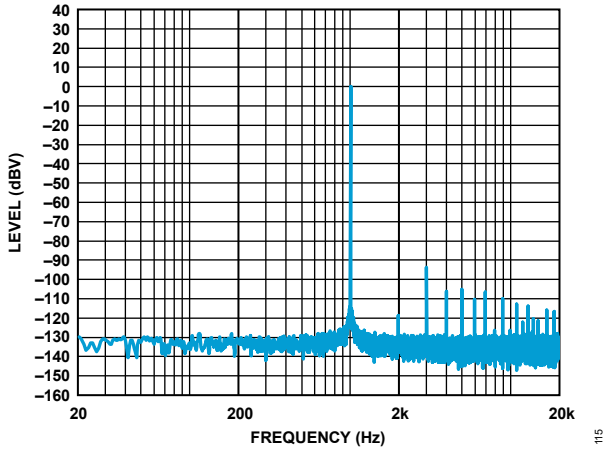

Figure 24. FFT, -1dBFS input, Fs = 48kHz, Signal Path = SDATAI_x to HPOUT, 16Ω Load

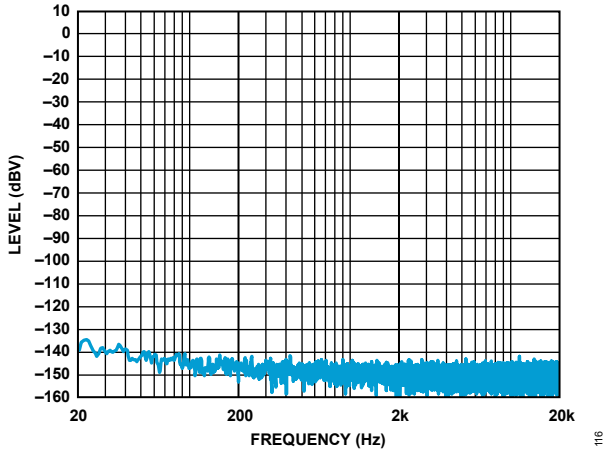

Figure 25. FFT, No Signal, Fs = 768kHz, Signal Path = SDATAI_x to FINT to FDSP to HOUT, 16Ω Load

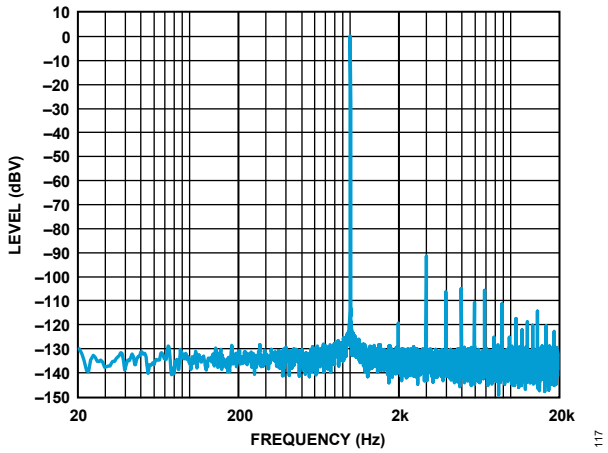

Figure 26. FFT, -1dBFS Input, Fs = 768kHz, Signal Path = SDATAI_x to FINT to FDSP to HOUT, 16Ω Load

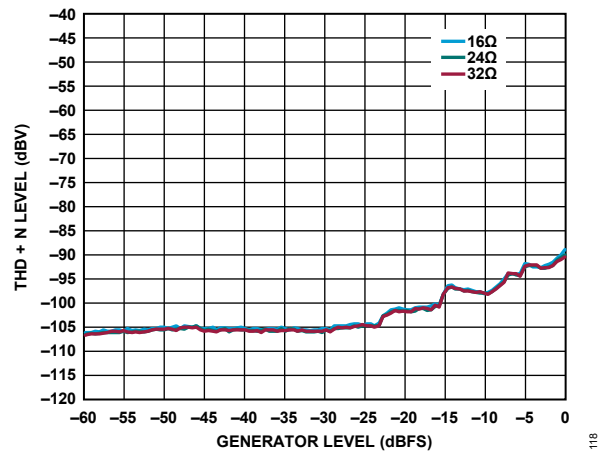

Figure 27. THD + N Level vs. Input Amplitude, Fs = 48kHz, Signal Path = SDATAI_x to HPOUT

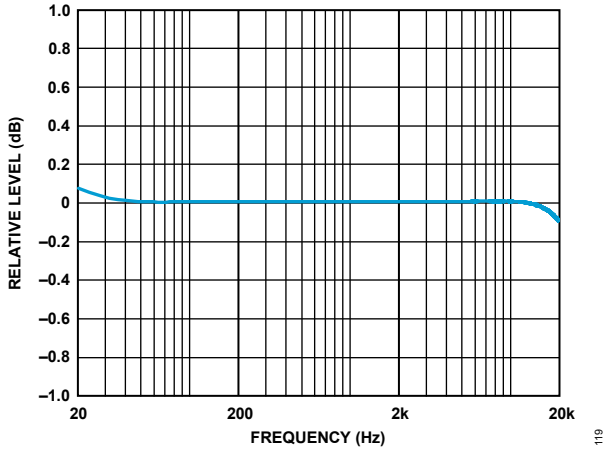

Figure 28. Relative Level vs. Frequency, Fs = 48kHz, Signal Path = SDATAI_x to HPOUT, 16Ω Load

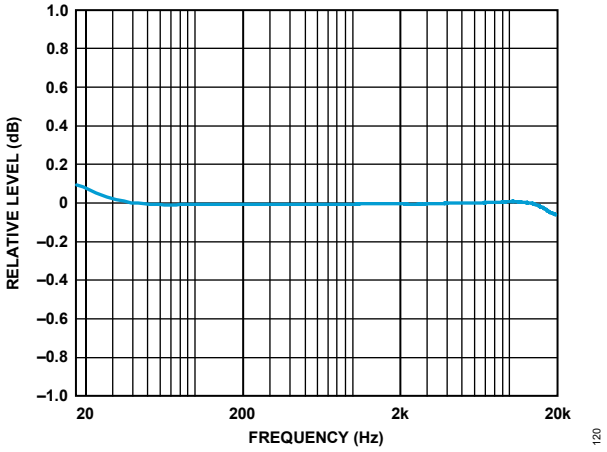

Figure 29. Relative Level vs. Frequency, Fs = 768kHz, Signal Path = SDATAI_x to FINT to FDSP to HPOUT, 16Ω Load

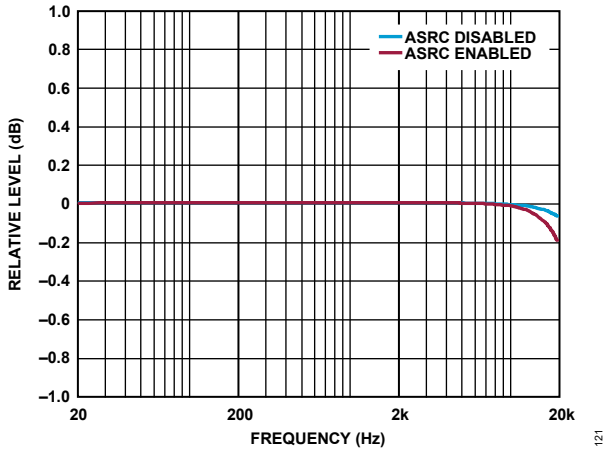

Figure 30. Relative Level vs. Frequency, Fs = 48kHz Except FDSP = 768kHz, Signal Path = SDATAI_x to ASRCI to FINT to FDSP to FDEC to ASRCO to SDATAO_x

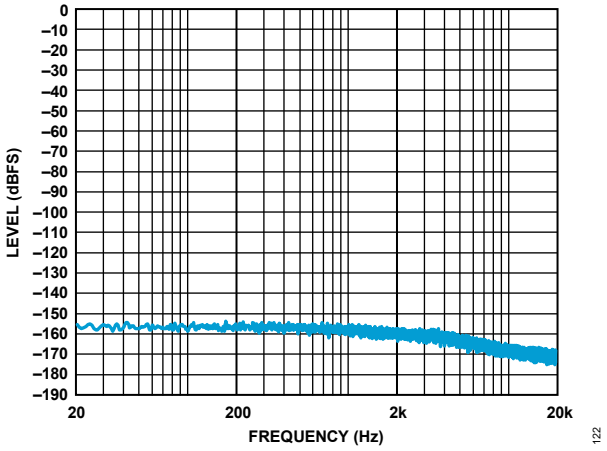

Figure 31. FFT, No Signal, Fs = 48kHz Except FDSP = 768kHz, Signal Path = SDATAI_x to ASRCI to FINT to FDSP to FDEC to ASRCO to SDATAO_x

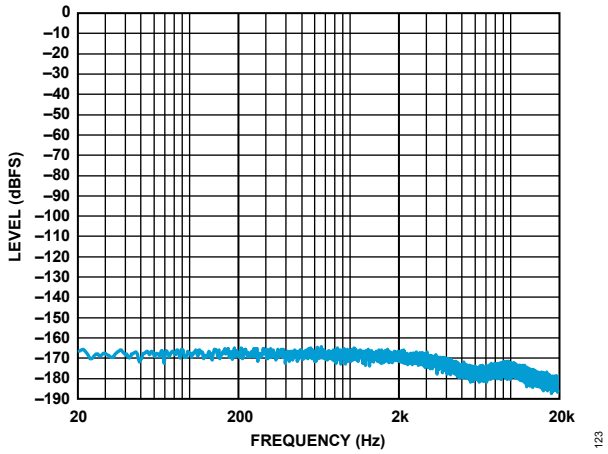

Figure 32. FFT, No Signal, Fs = 48kHz Except FDSP = 768kHz, Signal Path = SDATAI_x to FINT to FDSP to FDEC to SDATAO_x

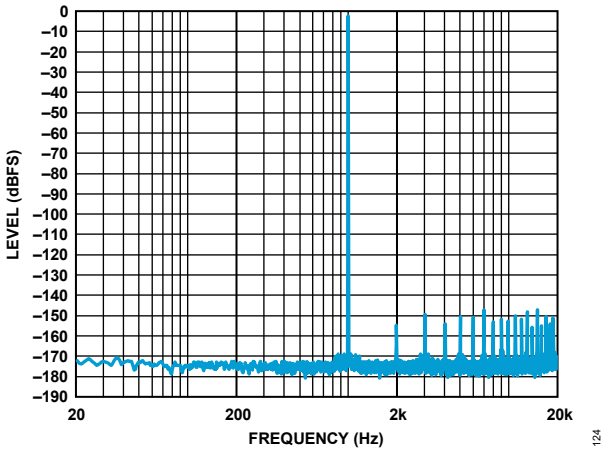

Figure 33. FFT, -1dBFS, Fs = 48kHz Except FDSP = 768kHz, Signal Path = SDATAI_x to ASRCI to FINT to FDSP to FDEC to ASRCO to SDATAO_x

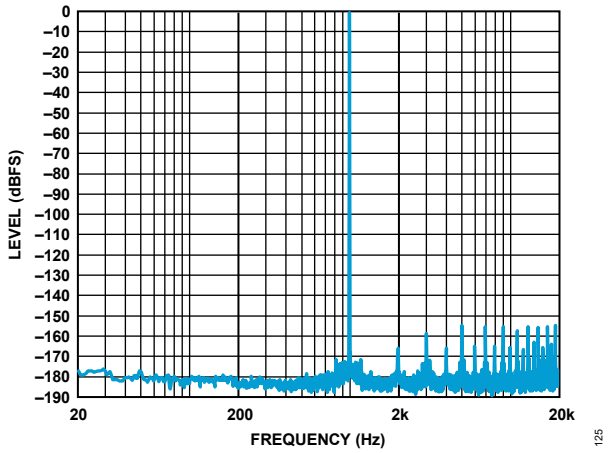

Figure 34. FFT, -1dBFS, Fs = 48kHz Except FDSP = 768kHz, Signal Path = SDATAI_x to FINT to FDSP to FDEC to SDATAO_x

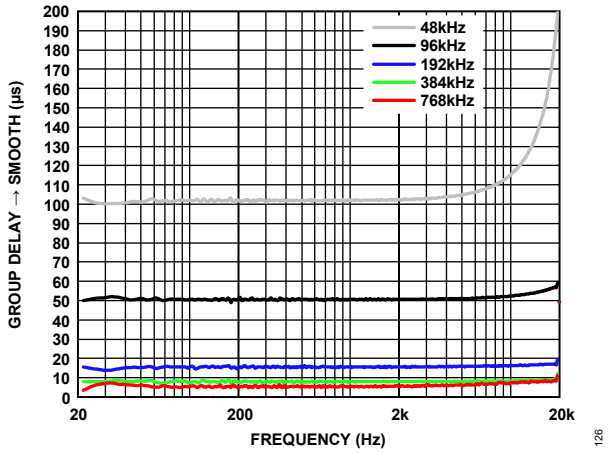

Figure 35. Group Delay vs. Frequency, Fs = 48kHz to 768kHz, Signal Path = AINx to FDSP to HPOUT

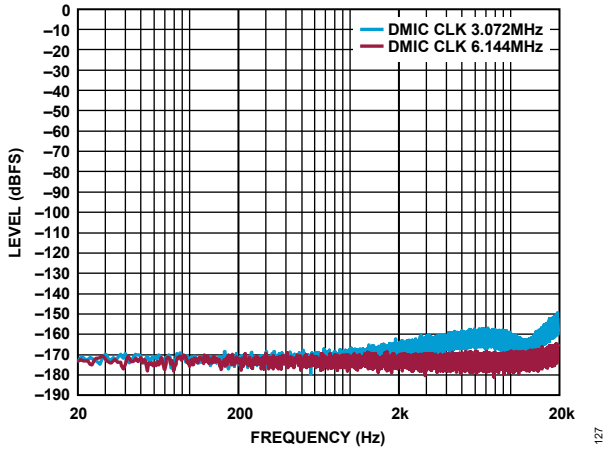

Figure 36. FFT, No Signal, Fs = 48kHz, Signal Path = DMICxx to SDATAO_x

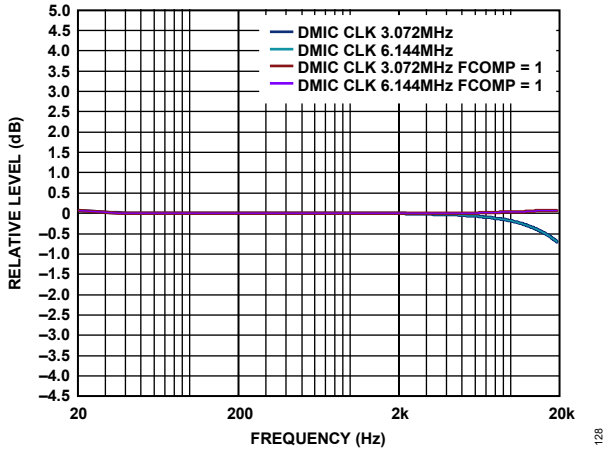

Figure 37. Relative Level vs. Frequency, Fs = 48kHz, Signal Path = DMICxx to SDTAO_x

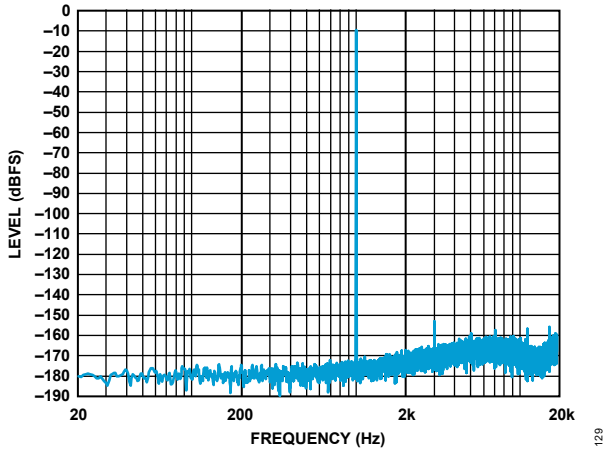

Figure 38. FFT, -10dBFS Input, DMIC_CLKx = 3.072MHz, Fs = 48kHz, Signal Path = DMICx to SDATAO_x

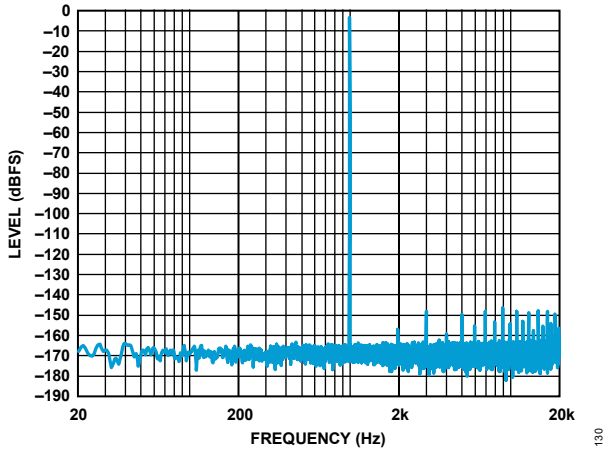

Figure 39. FFT, -10dBFS Input, DMIC_CLKx = 6.144MHz, Fs = 48kHz, Signal Path = DMICx to SDATAO_x

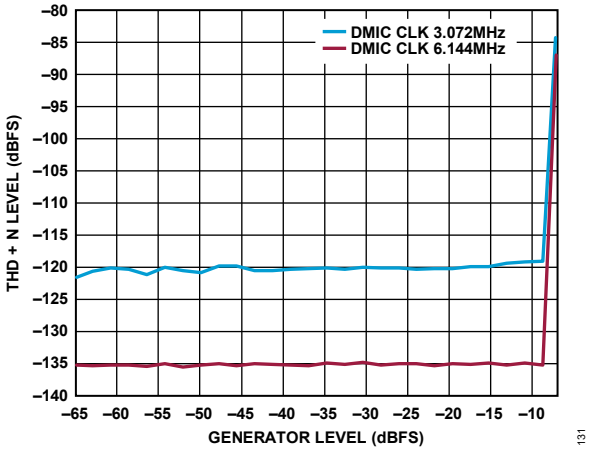

Figure 40. THD + N Level vs. Amplitude, -10dBFS, Fs = 48kHz, Signal Path = DMICxx to SDATAO_x

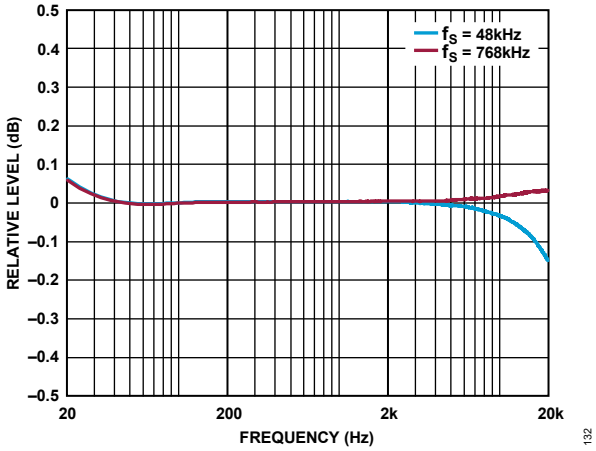

Figure 41. Relative Level vs. Frequency, Signal Path = AINx to HPOUT, 16Ω Load

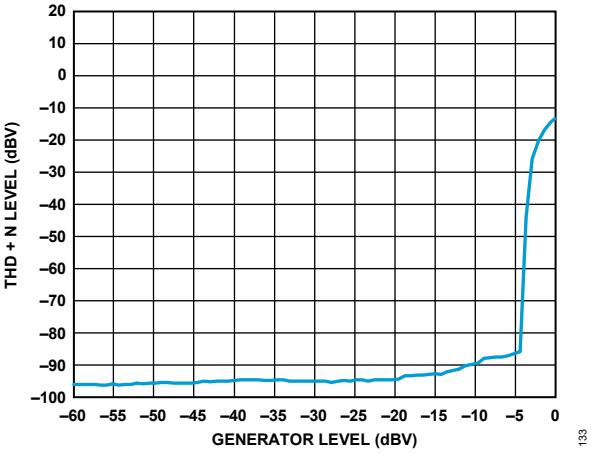

Figure 42. THD + N Level vs. Amplitude, Fs = 48kHz to 768kHz, Signal Path = AINx to HPOUT, 16Ω Load

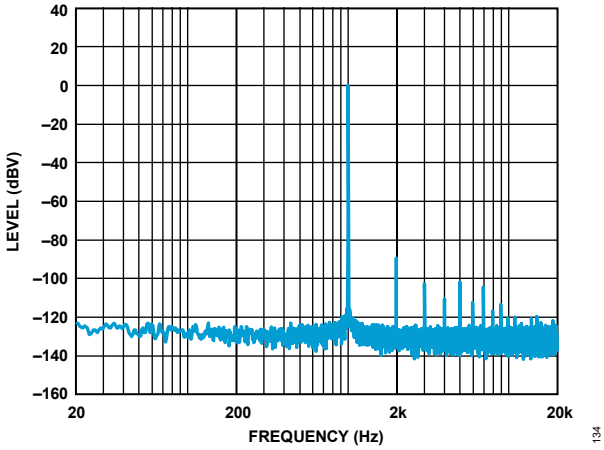

Figure 43. FFT, -7dBV Input, Fs = 48kHz to 768kHz, Signal Path = AINx to HPOUT, 16Ω Load

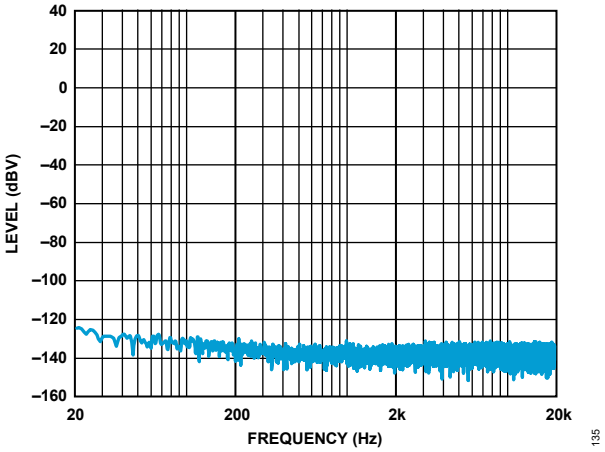

Figure 44. FFT, No Signal, Fs = 48kHz to 768kHz, Signal Path = AINx to HPOUT, 16Ω Load

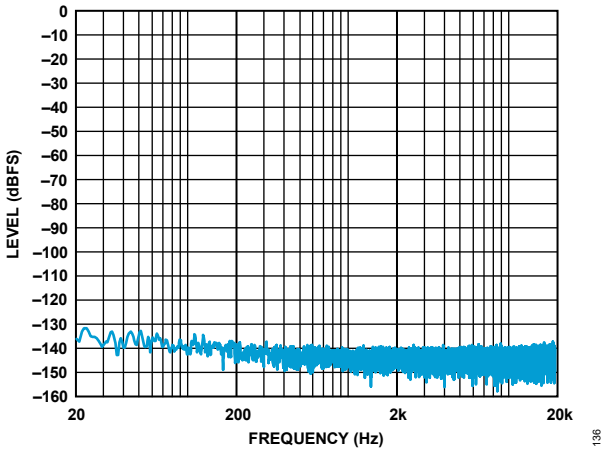

Figure 45. FFT, No Signal, Fs = 48kHz, Signal Path = AINx to SDATAO_x, No PGA (ADC Differential Mode)

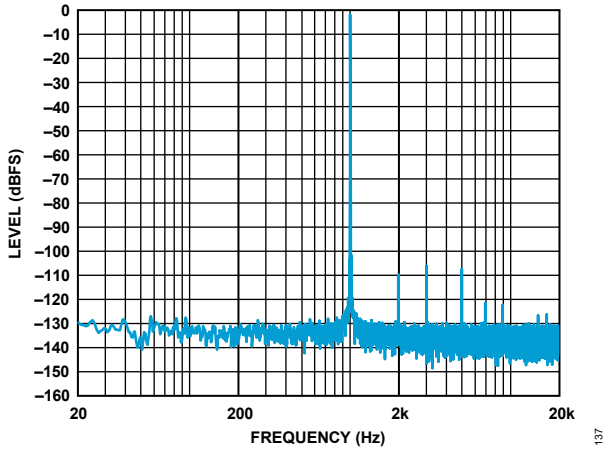

Figure 46. FFT, -1dBV Input, Fs = 48kHz, Signal Path = AINx to SDATAO_x, No PGA (ADC Differential Mode)

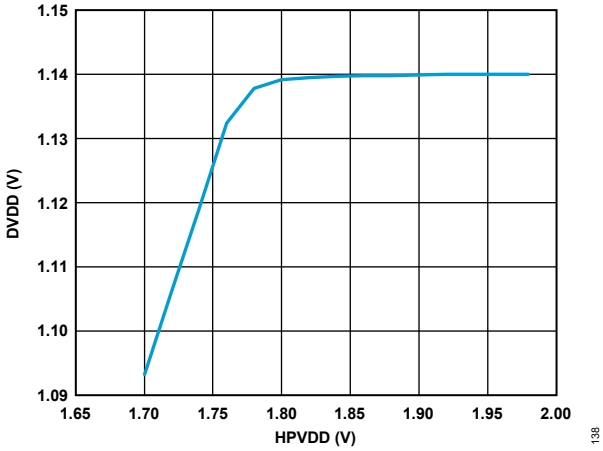

Figure 47. Integrated Switched-Cap DVDD Line Regulation vs. HPVDD Operating Range

SYSTEM BLOCK DIAGRAM

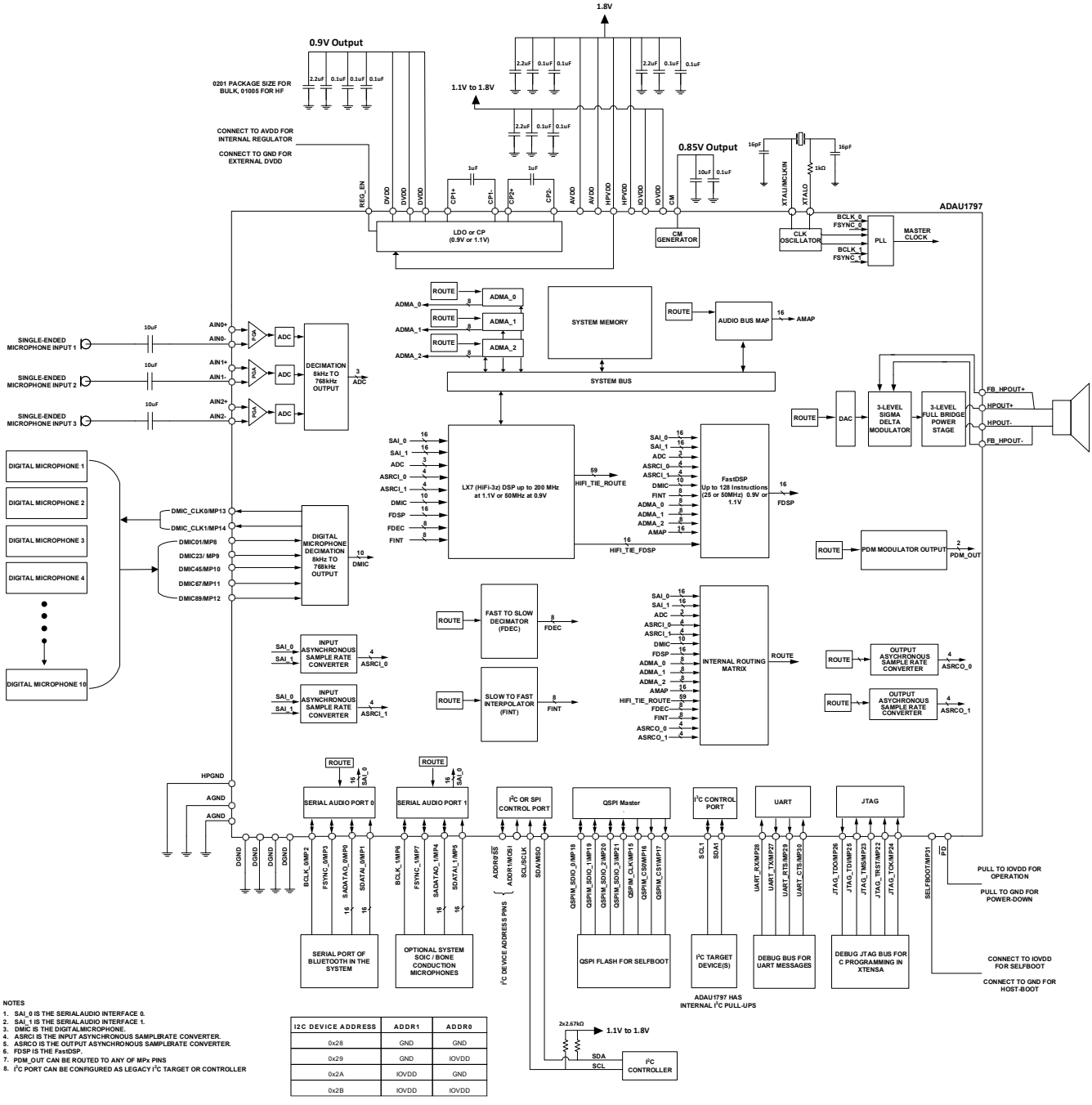

Figure 48. System Block Diagram (Analog Microphones, Self-Boot Mode)

THEORY OF OPERATION

The device is a low power, high-performance audio codec with dual audio processing cores. The device features both a low power HiFi 3z audio DSP core and a low latency optimized FastDSP core. The dual audio DSP cores paired with the low power, high-performance data converters make the device ideal for applications such as noise cancellation, transparency, personal sound amplification, and voice processing.

The device provides three low power, low-noise analog input channels each with a high-performance ADC. The flexible analog inputs accept both line-level and analog microphone input sources. Each analog input can be configured as a single-ended, differential, or pseudo-differential input with or without a programmable gain amplifier (PGA). In addition, the device also supports up to ten digital microphone input channels with two independent digital microphone output clock sources.

The analog output channel includes a high-performance DAC and a low-noise, high-efficiency differential headphone amplifier. The driver is an ultra-low power, pulse density, closed-loop, filter-less, sigma-delta Class-D amplifier that is capable of driving speakers with a 6Ω impedance or higher.

The input ADCs and output DAC are high-performance, 24-bit Σ-Δ converters that operate at a selectable 8 kHz to 768 kHz sampling rate. Each data converter channel includes a high resolution, digital soft volume control, and an optional high-pass filter with a configurable cutoff frequency.

The device provides two independent serial audio data ports. Each can be configured as either a master or slave interface, and they support I^2S, left justified, right justified, and up to 16 channel TDM compatible data formats. The data port supports either 1.2V or 1.8V logic levels (as set by IOVDD voltage).

The device also provides an I^2C/SPI compatible control port that supports both slave mode and master mode (with a software driver running in the HiFi 3z core) operation. In I^2C mode, the control port can be assigned one of 4 possible addresses and supports up to fast mode plus timing.

The Tensilica HiFi 3z core is optimized for low power audio processing. This core can either be directly programmed in C/C++ or graphically programmed using the *SigmaStudio®+* software from Analog Devices, Inc. It includes a library of configurable audio processing blocks such as filters, dynamics processors, mixers, and low-level DSP functions for fast, graphical development of custom signal flows. Software running on the core also enables access to additional hardware interface ports and operating modes including the I^3C, QSPI, and UART interface ports, master mode control for the I^2C/SPI port, and debug access through the JTAG interface port.

The FastDSP core has a reduced instruction set optimized for latency-critical applications such as noise cancellation and ambient transparency. The program and parameter random access memory (RAM) can be loaded with a custom audio processing signal flow built using *SigmaStudio+*.

The device also has a self-boot function that can load the program and data/parameter RAMs of both cores along with the register settings for the device on power-up using an external electronically erasable programmable read-only memory (EEPROM) or flash memory over a master quad SPI interface. The external flash memory is fully memory-mapped to the HiFi 3z bus fabric.

The flexible *SigmaStudio+* software connects to the device control port and can both configure the device registers, as well as program and control the cores. The graphical user interface (GUI) provides the flexibility and programmability needed by an experienced DSP programmer. However, the GUI is also designed to be user-friendly to enable even less experienced users with baseline digital or analog audio processing knowledge to design the DSP signal flow and export the flow to a target application. A full description of the device register map and memory maps is available in the companion programming guide.

In *SigmaStudio+*, the user can connect graphical blocks (such as biquad filters, volume controls, and arithmetic operations), compile the design, and load the program and parameter files into the device memory through the control port. The software also allows the user to download the design to an external EEPROM or Flash to prepare for self-boot operation.

The device features an on-board, bypassable fractional PLL that can generate all required internal clocks. The external clock input can either be derived from the master clock input or the bit clock input of either serial audio data port. If the input source clock frequency is within the 900kHz to 2.1MHz range it can be directly accepted by the PLL. Higher source clock frequencies up to 49.152MHz can also be accepted by first dividing the input clock down into this range by using the configurable input pre-scaler divider. For standalone operation, the source clock can be generated using the on-board crystal oscillator.

The device is available in a small, 77-ball, 3.24mm × 4.83mm WLCSP (0.4mm pitch and 0.5mm height) and is specified for operation over the extended -40°C to +85°C temperature range.

Signal Routing Diagram

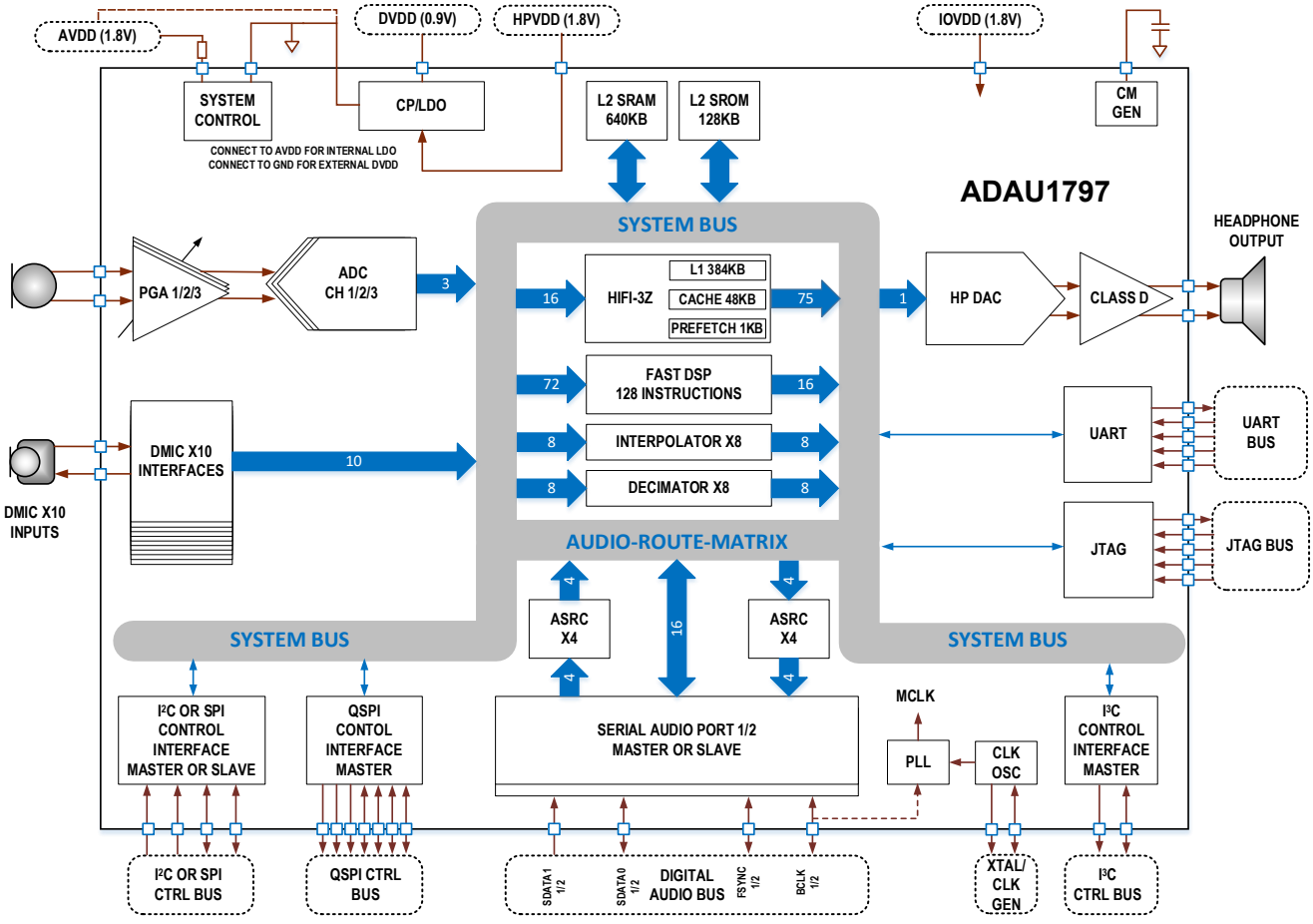

Figure 49. Device Input and Output Signal Routing

Power Supplies and Sequencing Requirements

The device requires four primary power supplies. Three of these supplies must be provided externally to the device. These are the primary analog supply (AVDD), the headphone amplifier and internal linear regulator/charge pump supply (HPVDD), and the digital interface logic level supply (IOVDD). The core digital supply (DVDD) can either be provided externally or generated internally from the HPVDD supply.

The analog supply (AVDD) and headphone/internal regulator supply (HPVDD) support operation from a 1.8V nominal level. The digital interface logic supply (IOVDD) supports operation from either a 1.8V or 1.2V nominal level. The required digital core supply level (DVDD) is dependent on the required core processing load of the implemented use case. In low power/light processing load cases, a 0.9V nominal DVDD supply level is acceptable (HIFI_SPEED = 0), while high performance/heavy processing load cases require a 1.1V nominal DVDD supply level (HIFI_SPEED = 1). This applies both for an internally generated or an externally provided DVDD source.

Power Supply Sequencing

During the initial system power-up supply sequencing, the AVDD and HPVDD supplies must be powered up and within their operating range before or at the same time that IOVDD is powered up into its operating range. Do not power up the IOVDD supply when the AVDD and HPVDD supplies are outside of their normal operating ranges.

If the DVDD supply is to be generated by the internal linear regulator or charge pump, then the REG_EN input should be externally connected to the HPVDD supply. Otherwise, the REG_EN input must be connected to ground and the DVDD supply must be externally supplied to the device before power-up.

Before powering down the device supplies, all signal input and output channels should be muted or powered down. Audible glitches may be recorded or played back if any supply is removed while audio channels are active.

Hardware Full Chip Power-Down States

The device is placed into the hardware full chip power-down (or hardware shutdown) state when the $\overline{PD}$ input pin is asserted low. This is the lowest power device state, and in this state, external supplies can either be present or fully powered down (with proper sequencing). In this state, the internal regulators for DVDD (if used) are also disabled and powered down. The device is fully reset in this state and retains no state memory (all registers and blocks return to their PoR states/configurations). No interface communication with the device is possible in this state.

As with the supply sequencing, before entering the hardware power-down state (asserting the $\overline{PD}$ input low), all signal input and output channels should be muted or already powered down. Otherwise, audible glitches may be recorded or played back before the hardware shutdown sequence is completed.

When the device transitions out of reset and the hardware power-down state (supply sequencing is complete and the $\overline{PD}$ input is asserted high), the device enters the software full chip power-down (software shutdown) state.

Software Full Chip Power-Down State

The software power-down state is the lowest power state where software/driver control is possible, and the control register map retains any programmed settings. In this state, the I^2C/SPI control ports operate as slave interfaces. Most other blocks are powered down except for internal DVDD regulators (if the REG_EN input is high), and both the common mode voltage generator and the crystal oscillator (by default).

In this state the device power enable bit (POWER_EN) defaults to 0. To transition out of the software power-down state set POWER_EN = 1. Conversely, from a power-up state, setting POWER_EN = 0 transitions the device back into the software power-down state. As with previous power-down scenarios, to avoid audible glitches all signal input and output channels should be muted or powered down before setting POWER_EN = 0.

By default, in the software power-down state the common-mode output voltage (and CM output pin) is active (CM_KEEP_ALIVE = 1). To reduce power consumption, this can be disabled by setting CM_KEEP_ALIVE = 0. However, as a result, the device will have a longer power-up time as this adds a wait time of up to 35ms (maximum) for the common mode voltage to charge before any analog blocks (such as the PLL or converters) can be enabled. Conversely, with CM_KEEP_ALIVE = 1, the software power-down current is increased but the power-up time is faster as this wait time can be omitted.

Similarly, the internal digital clocks can be gated to reduce power consumption when idle in the software power-down state by setting XTAL_EN = 0. To idle the device in the lowest possible power configuration for the software power-down state (POWER_EN = 0), set both CM_KEEP_ALIVE = 0 and XTAL_EN bit = 0.

The digital portion of the chip has several power domains. By default, in the software power-down state, only the domain that powers the control ports and their associated registers are powered on. The remainder of the digital domain has its power supplies gated resulting in the loss of the internal states. In addition, to retain core (FDSP/HIFI 3z) state memory in the software power-down state set the KEEP_MEM bit to 1.

Device Power-Up Sequencing

The device transitions out of the software power-down state when POWER_EN is set to 1. The power supplies on the rest of the digital portion of the chip are not enabled until this occurs, so this must be set first during the power-up sequence. After the internal digital power supplies are powered up, the PLL is locked, and other needed sequencing is complete, the POWER_UP_COMPLETE bit sets to 1 and the DSP memories can be programmed or accessed.

Interrupt requests (IRQs) can also be sent when the power-up is completed. If the IRQs are used to indicate power up is complete, then the appropriate IRQs must be unmasked. For the power-up sequence, these are the IRQ1_POWER_UP_COMPLETE and IRQ2_POWER_UP_COMPLETE bits. The associated mask bits that must be cleared (they are set by default) are the IRQ1_POWER_ UP_COMPLETE_MASK and IRQ2_POWER_UP_COMPLETE bits.

The device has highly flexible block-level power controls. Each channel of each block can be powered on or off separately. There is a control bit, MASTER_BLOCK_EN, that by default is 0 and that overrides all block level enables except for PLL_EN, XTAL_EN, HIFI_EN, and FDSP_EN. The PLL, HiFi 3z DSP, and FastDSP can be enabled even when MASTER_ BLOCK_EN = 0. All other blocks are always in power-down in this state, allowing the PLL to be enabled and locked and the DSP memories to be initialized before all other signal path blocks are enabled.

When configuring the device, it is recommended to fully set up all control registers and block level power controls to their desired state before setting MASTER_BLOCK_EN = 1. This allows the PLL to lock and initialize the DSP memories to be ready to be used before MASTER_BLOCK_EN is set.

Block-level power controls and other settings can be changed on the fly while the chip is active. However, care must be taken when enabling or disabling blocks other than the DAC and/or headphone mode blocks that are actively routed out to the DAC and/or headphone mode as audible glitches may occur.

To power down the chip, disable or mute any active audio channels, set MASTER_BLOCK_EN low, and then set POWER_EN low. This allows the device to power down all blocks with any required power-down sequencing.

Table 8. General Power-Up Sequencing Overview

STEP	DESCRIPTION
1	Set $\overline{PD}$ = 1 to exit reset and the hardware power-down state, and to transition the device to the software power-down state. This enables the internal DVDD regulator (if REG_EN is asserted high), the slave I^2C/SPI ports, and enables the common mode voltage generator and crystal oscillator.
2	Wait 15ms if the REG_EN input pin was asserted high in step 1.
3	Set POWER_EN = 1 to exit the software power-down state. This ungates the digital power domains and initiates the power-up sequence. The POWER_UP_COMPLETE bit sets when this is complete. Note: XTAL_EN is set to 1 by default in this sequence and must be set this way to complete this step.
4	Configure the PLL using CLK_CTRLx registers and set the XTAL_MODE and PLL_EN bits.
5	Configure all other setup bits for the desired use case while the PLL is locking. This can be done at any other time after $\overline{PD}$ = 1 but should be completed before step 8.
6	Ensure that all digital power domains are finished powering up, the PLL is locked, and the power-up sequencing is complete. PLL lock is verified when the PLL_LOCK bit is 1, and power-up sequencing is verified when the POWER_UP_COMPLETE bit is 1. Wait until both bits are set.
7	Ensure that HIFI_EN = 1 and FDSP_EN = 1 if either core is required for the desired use case, and initialize the static RAMs (SRAMs).
8	Set MASTER_BLOCK_EN = 1 to power up the blocks that are enabled for the use case.
9	Set FDSP_RUN = 1 and HIFI_RUN = 1 to instruct the DSPs to operate.

Power-Up Sequencing Use Case Example: ADC Record Input to Headphone Amp Output

To illustrate the power-on sequencing for a real use case configuration, a full example sequence of supply application, device register writes, and required wait times follows. The sequence configures the device for a basic use case (analog input to ADC channel 0 to the playback DAC to the headphone amplifier output), and this case assumes an MCLK input of 24.576 MHz and that an internally generated 0.9V DVDD level is used. The sequence is executed in host boot mode.

While this sequence does not utilize the serial audio data ports or program the FDSP or the HiFi 3z cores, it does include steps that illustrate the proper sequencing location for the data port, processor enables, and RUN bits for configuring the input and output audio data converter channels.

NOTE: The data writes in this sequence are endian-swapped, meaning that for each register the last byte (bits[7:0]) is shown first, and the first byte (bits[32:25]) is shown last. The bits are ordered from MSB first to LSB last.

Table 9. Use Case Power-Up Sequencing Example

STEP	ACTION	DESCRIPTION
1	Power-Up External AVDD/HPVDD/IOVDD	Power up the external HPVDD and AVDD supplies to within their operating ranges first, then power up the external IOVDD supply to its operating range.
2	Power-Up DVDD	Verify that the REG_EN input pin is pulled high (HPVDD) and wait 15ms for DVDD LDO regulation to Settle.
3	Verify CM Fast Charge is Enabled	If CM_KEEP_ALIVE = 0 and REG_EN = 0, ensure that CM_STARTUP_OVER = 0 (located in Register Address 0xF000002C).
4	Set Power Enable	Set POWER_EN = 1 by writing 0x01000000 to Register Address 0xF000003C.
5	Wait for CM Voltage	If CM_KEEP_ALIVE = 0 and REG_EN = 0, wait 35ms.
6	Disable CM Fast Charge	Set CM_STARTUP_OVER = 1 by writing 0x02000000 to address 0xF000002C.
7	Apply Clocks	Apply the necessary clocks to source the PLL (MCLK or BCLK0/1).

8	Disable Clock Gating	Set XTAL_EN = 1 by writing 0x02000000 to Address 0xF0000030. XTAL_EN must = 1 to ungate the clock tree, even if the crystal oscillator is not used.
9	Clock Configuration	Configure XTAL_MODE (crystal oscillator used), set the PLL for Integer Mode and set MCLK as the PLL input source by writing 0x08070000 to Address 0xF0000040.
10	Set Clock Ratio	Set PLL_INPUT_PRESCALER = 16 by writing 0x10000000 to Address 0xF0000044.
11	Set Clock Ratio	Set PLL_INTEGER_DIVIDER = 128 by writing 0x80000000 to Address 0xF0000048. Note these PLL settings configure the PLL output for 196.608MHz.
12	Update PLL Settings	Set PLL_UPDATE = 1 to ensure these changes take effect in the PLL by writing 0x03000000 to Address 0xF0000030.
13	Verify PLL Lock	Check PLL_LOCK (read 0xF0000408). If PLL_LOCK = 0, wait until PLL_LOCK = 1.
14	Verify Successful Power-Up Complete	Check POWER_UP_COMPLETE = 1 (read 0xF000040C). If POWER_UP_COMPLETE = 0, wait until POWER_UP_COMPLETE = 1. If POWER_UP_COMPLETE = 1 and PLL_LOCK = 1, continue configuring the device.
15	Enable Boot Loader	Set PROC_EN = 1 by writing 0x10000000 to Address 0xF0000034. Note that an app pack is not loaded into the L2 memory in this example, but the boot loader must still be enabled to ensure access to the HiFi 3z memory, FDSP memory, or Main Map registers. PROC_EN can be cleared at any subsequent point in the sequence once the boot loader is enabled.
15	Enable Switched Cap Regulator	Enable the switched-cap regulator to switch over from the LDO regulator, and set the generated DVDD level to 0.9V. Set CP_EN = 1 and SEL_0V9 = 1 by writing 0x32000000 to Address 0xF000002C.
16	Set ADC Sample Rate	Set the ADCs to f_S = 48kHz by writing 0x04040000 to Address 0xF0000058.
17	Set Differential Analog Input Mode	Set the ADCs to differential mode by writing 0x07000000 to Address 0xF000005C.
18	Set DAC Sample Rate	Set the DAC_FS = 48kHz by writing 0x44000000 to Address 0xF00000F4.
19	Route ADC to DAC	Set DAC_ROUTE0 = ADC0 by writing 0x38000000 to Address 0xF0000108.
20	Enable Converters	Power up ADCs and playback path (write 0x17000000 to Address 0xF000000C).
21	Serial Port Setup when Required	Ensure all necessary clocks are now present on the serial port(s) if needed for a different concurrent signal channel use case.
22	Core Configuration When Required	Set PROC_BYPASS = 0, PROC_ EN = 0 and FDSP_EN = 1 by writing 0x01000000 to Address 0xF0000034. (**Note:** Insert other core configuration if needed)
23	Set Master Block Enable	Set MASTER_BLOCK_EN = 1 by writing 0x33000000 to Address 0xF000002C.
24	Set HiFi 3z Core Enable (If Used)	If using V_{DVDD} = 0.9V, set HIFI_SPEED = 0 (write 0x01000000 to address 0xF00001FC), then set PROC_EN = 1 (write 0x10000000 to Address 0xF0000034).
25	Set FDSP Core Run Bit (If Used)	Set FDSP_RUN by writing 0x01000000 to Address 0xF00001B8. If both HiFi 3z and FDSP are used write 0x11000000 instead.
26	Verify data Input	Ensure that audio data is now present if using the serial audio data port input.
27	Enable ADC Soft Volume Ramping	Set ADC_HARD_VOL = 0 by writing 0x40000000 to Address 0xF000006C.
28	Enable DAC Soft Volume Ramping	Set DAC_HARD_VOL = 0 by writing 0x44000000 to Address 0xF00000F8.
29	Unmute ADCs	Unmute ADCs by writing 0x00000000 to Address 0xF0000074.
30	Unmute DAC	Unmute the DAC and playback amp (write 0x04000000 to Address 0xF00000F8).
31	Check Amp Output	Verify audio signal output from the headphone amplifier.

Internal DVDD Linear Regulator (LDO) and Switched-Cap Regulator

The device features both a fully integrated linear regulator and an internal switched-cap regulator. These can be used to internally generate the required 0.9V or 1.1V nominal DVDD voltage from the HPVDD input (nominal 1.8V).

If the REG_EN input pin is tied to ground when the device enters the software power-down state, the internal regulators are disabled, and an appropriate DVDD voltage must be supplied externally to the DVDD pin.

If the REG_EN pin is instead tied to HPVDD, then when the device enters the software power-down state the internal LDO regulator automatically enables and generates the required DVDD voltage (maximum settling time of 20ms).

Once the device is in the software power-down state (if the internal DVDD regulator is enabled), the integrated switched-cap regulator (charge pump) can be enabled instead of the LDO by setting the CP_EN bit to 1. To utilize this integrated regulator mode, two additional external caps are required and must be populated. The switched-cap regulator provides a more efficient power structure than the LDO for DVDD voltage regulation, however, two additional external caps are required (and must be populated) and line regulation is slightly degraded.

When using internal DVDD regulator mode, the SEL_0V9 bit determines the DVDD voltage output level and by default, the output is set to 1.1V. It can also be configured to 0.9V (SEL_0V9 = 1), however, this should only be selected when the HiFi 3z core is running at 50MHz (HIFI_SPEED = 0). The internal DVDD voltage regulation level should be selected during device initialization before activating any configured signal channels.

If the DVDD regulation level also needs to be changed for different use cases, to avoid any potentially audible glitches it is recommended that all active audio channels should first be muted (or disabled if not used in the new use case). Next, if the HiFi 3z core or FDSP is active, then PROC_EN and FDSP_RUN should be cleared before changing the DVDD voltage. Once this is done, the new DVDD regulation level can be selected (with the SEL_0V9 bit). The system should then wait for the DVDD regulation settling time to elapse (maximum of 15ms). During this time, the software should not read from the device's memory. Once the new DVDD regulation level is stable, the new use case and applicable processors can be fully enabled, and all audio channels for the new use case can be unmuted and/or enabled.

The internal regulators require the CM voltage to be powered up to operate. Therefore, when CM_KEEP_ALIVE = 1, both the CM output and internal DVDD regulators remain powered up in the software power-down state. To minimize device power-down state power consumption when using the internal regulator, either disable common mode keep alive (CM_KEEP_ALIVE = 0) or place the device into the hardware power-down state.

Clock Configuration and Control

The device requires a valid external reference clock source with a frequency between 0.9MHz and 50MHz. Using the PLL_SOURCE bits, the device can be configured to accept this clock from either the crystal/master clock input pin (XTALI/MCLKIN) or from either of the bit clock input pins (BCLK_x).

If the external clock source is a crystal oscillator, the terminals should be connected between the XTALI/MCLKIN and XTALO pins as illustrated in the *System Block Diagram*. The internal crystal amplifier should then be selected and enabled by setting the XTAL_MODE bit to 1 (default). If instead a line-level master clock signal is used, the crystal amplifier can be disabled by setting XTAL_MODE to 0.

The XTAL_EN bit is used to gate all internal clocks (minimizes idle power in the software power-down state). For all cases (bit clock, master clock, or crystal clock input), XTAL_EN must be set high for device power-up to be possible.

PLL Overview

The integrated PLL is enabled with the PLL_EN bit and uses the selected input reference clock source (set with the PLL_SOURCE bits) to generate all required internal reference clocks and any master mode interface output clocks.

The PLL accepts input clock frequencies between 0.9MHz and 2.1MHz, and before reaching the PLL the selected external input clock signal is routed through an integer clock prescaler divider. The divide ratio must be set (with the PLL_INPUT_PRESCALER bits) to divide down the input source clock frequency to within this range (see below).

$$f_{PLL\_INPUT\_CLOCK} = f_{EXTERNAL\_INPUT\_CLOCK}/(PLL\_INPUT\_PRESCALER)$$

The PLL supports both an integer and fractional clock ratio mode (for PLL input to output clock ratio), and this is selected with the PLL_TYPE bit. In either mode, the PLL feedback ratio must be configured such that the PLL output frequency is fixed at 196.608MHz. All specified internal sample rates within the documentation assume this PLL output frequency, which is a 48kHz sample rate x 4096. The clock and PLL structure are illustrated in *Figure 50*.

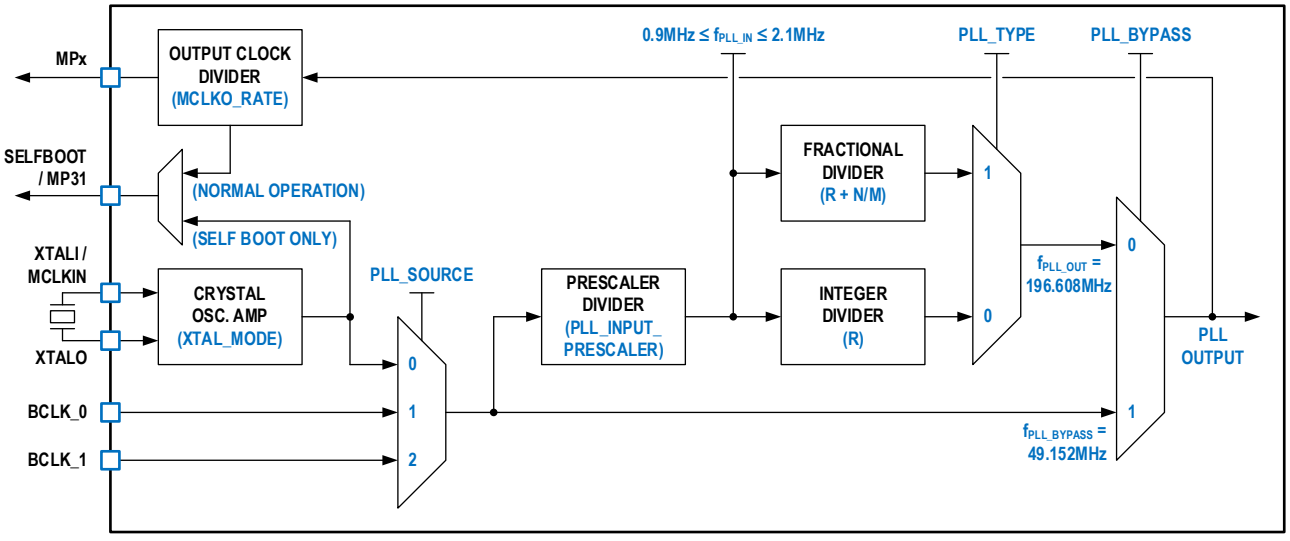

Figure 50. Clock Input and PLL Block Diagram

PLL Integer Clock Ratio Mode

Integer clock ratio mode (PLL_TYPE = 0) can be used when the target PLL output frequency of 196.608MHz is an integer multiple of the PLL input clock frequency (after prescaler divide). The integer feedback clock divider calculation is shown below, and the divider ratio value R is set with the PLL_INTEGER_DIVIDER bits.

$$f_{PLL\_OUTPUT\_CLOCK}/f_{PLL\_INPUT\_CLOCK} = R$$

For example, if the selected external reference clock input frequency is 24.576MHz, and the integer clock prescaler divider ratio is set to 16 (PLL_INPUT_PRESCALER = 16), the PLL input clock frequency will then be:

$$f_{PLL\_INPUT\_CLOCK} = f_{EXTERNAL\_INPUT\_CLOCK}/(PLL\_INPUT\_PRESCALER)$$

$$1.536MHz = 24.576MHz/16$$

The required PLL output frequency is 196.608MHz, so the integer feedback clock ratio R must be 128 (set with the PLL_INTEGER_DIVIDE bits). In integer clock ratio mode, the fractional mode numerator and denominator settings have no effect (M and N as set by the PLL_DENOMINATOR and PLL_NUMERATOR bits respectively)

$$f_{PLL\_OUTPUT\_CLOCK}/f_{PLL\_INPUT\_CLOCK} = R$$

$$196.608MHz/1.536MHz = 128$$

By default, the power on reset states of the prescaler and integer divider bits are configured for a typical use case with a 24.576MHz input reference clock source to the PLL.

PLL Fractional Clock Ratio Mode

Fractional clock ratio mode (PLL_TYPE = 1) must be used when the target PLL output frequency of 196.608MHz is NOT a simple integer multiple of the PLL input clock frequency (after prescaler divide). Instead, a mixed clock ratio calculation is used (as shown below). The integer ratio portion R is still set with PLL_INTEGER_DIVIDER, and the fractional clock denominator M and numerator N are set with the PLL_DENOMINATOR and PLL_NUMERATOR bits.

$$f_{PLL\_OUTPUT\_CLOCK}/f_{PLL\_INPUT\_CLOCK} = R + N/M$$

For example, if the selected external reference clock input frequency is 13MHz then one possible integer clock prescaler divider ratio is 10 (PLL_INPUT_PRESCALER + 1 = 10) resulting in a PLL input clock frequency of 1.3MHz.

$$196.608MHz/1.3MHz = 151.2369 = 151 + 77/325$$

Taking the ratio of the PLL output clock to the input clock results in a value of ~151.2369. Splitting this into whole numbers and fractional portions results in 151 + 77/325 or R = 151, N = 77, and M = 325. Note that when the PLL is used in fractional mode, it is important that the N/M ratio be kept within the 0.1 ≤ N/M ≤ 0.9 range to ensure correct operation of the PLL. When used in fractional mode, the input to the PLL after the input divider must be ≥1MHz.

PLL Reference Clock Output

If the selected external PLL clock source is from the XTALI/MCLKIN input (PLL_SOURCE = 0x00), the device can also output this clock or a divided-down version of it to any available multipurpose pin. The input clock (XTALI/MCLKIN input) to output clock ratio (on a multipurpose output) is selected with the MCLKO_RATE bits and can be set to no divider (divide by 1) or to any power of 2 integer divider ratio from 2 to 128.

By default, the SELFBOOT/MP31 pin outputs the XTALI/MCLKIN input clock divided by 2. This occurs automatically when the device completes the transition from the hardware power-down state to the software power-down state. If this clock is not needed, to reduce power consumption it can be disabled with the MP31_MODE bits.

PLL Bypass Mode Operation

The device also supports operation with the PLL bypassed (PLL_BYPASS = 1). This is only possible if the selected external clock source (PLL_SOURCE) is at a fixed 49.152MHz frequency.

Power consumption is reduced in PLL bypass mode. All internal blocks operate the same except for the HiFi 3z DSP core. In PLL bypass mode, the HiFi 3z DSP core can only be set to run at the slower 49.152MHz speed setting (HIFI_SPEED = 0), and as a result, the number of instructions it can execute is limited (≤ 50MIPs).

Table 10. PLL Bypass Mode Power Reduction

PLL_BYPASS	PLL OPERATION	RELATIVE POWER CONSUMPTION (mW)
0	Used	+0.968
1	Bypassed	—

If the 49.152MHz external clock is available, then there is no downside to operating in PLL Bypass mode other than the HiFi 3z DSP core processing limitation.

PLL Multichip Clock Phase Synchronization

If multiple devices are operating from the same shared external reference clocks, then the internal audio channels of the devices can be phase synchronized. The synchronization clock source is selected by setting the SYNC_SOURCE bit, and it must be set to select the same clock source on each device (to be synchronized).

If the shared input frame clock (FSYNC_x from either port) is synchronous to the PLL-derived core clock, then the same frame clock input can be selected as the sync source for each device. However, if the shared frame clock is asynchronous to the core clock, then the sync source for each device must instead be set to one of the input asynchronous sample rate converters (ASRCI_x). If no serial audio data ports are used but each device uses the same external clock source (to the PLL), then an internal PLL-derived sync source can be used instead.

PLL Clock Configuration Sequencing

To configure the PLL during initialization or to reconfigure it for a different use case, the following sequence is recommended. To avoid potentially audible glitches, the PLL should not be reconfigured while it is active.

1. Ensure that POWER_EN = 1.
2. Ensure that PLL_EN = 0.
3. Configure (or reconfigure) the PLL control bits.
4. Write 1 to PLL_UPDATE to propagate the PLL settings.
5. Enable (or re-enable) the PLL by setting PLL_EN = 1.

Other blocks can be powered up while the PLL is not enabled or locked. However, if the PLL is enabled and not locked, all other circuitry waits until the PLL is locked to begin the power-up sequences.

All hardware control registers can be accessed at any time during PLL initialization, before PLL is enabled, or during PLL lock. However, to access the HiFi 3z DSP or FDSP memories, the respective core must be enabled (HIFI_EN = 1 or FDSP_EN = 1 respectively), and the PLL (if not bypassed) must be locked.

Input Signal Channels

The device provides three analog input signal channels each with a low power, low noise ADC. In addition, the device also accepts up to ten digital microphone input sources.

Analog Input Channels

The three flexible analog input channels accept both line-level and analog microphone input sources. Each analog input channel individually supports either a single-ended, differential, or pseudo-differential configuration, with or without a programmable gain amplifier (PGA).

When enabled (ADCx_EN), the analog inputs of each channel (AINx- and AINx+) are internally biased to the input common-mode voltage (V_{CM}). For external reference, the common mode voltage is output on the CM pin. The simplified block diagram is shown in *Figure 51*.

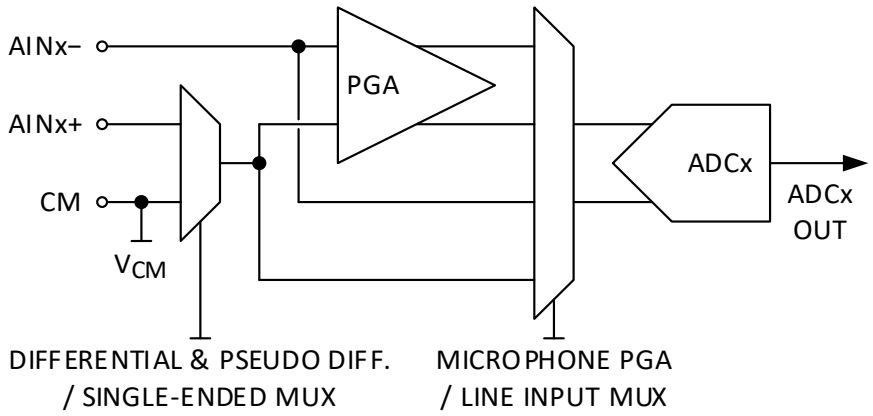

Figure 51. Analog Input Channel Simplified Block Diagram

Analog Input Mode Configuration

The analog input mode (single-ended, differential, or pseudo-differential mode) is selected for each channel using the AFEx_DIFF and AFEx_PDIFF bits. Each of these input modes can be used with the series programmable gain amplifier (PGA) either enabled or disabled (and bypassed) using the PGAx_EN bits. The default setting (after power on or reset) is single-ended mode with PGA disabled.

When a given analog input channel is configured for single-ended (SE) mode, the input signal must be applied to the negative analog input pin (AINx-). As a result, for these configurations, the input signal polarity is inverted and can be corrected by setting the ADCx_INVERT bits.

In differential (DIFF) mode, the input signal can be applied (with correct polarity) to both the positive (AINx+) and negative (AINx-) analog input pins.

In pseudo-differential (PDIFF) mode, the analog input signal can be applied to either the positive (AINx+) or negative (AINx-) input (with the other input AC coupled to the ground reference of the signal source). When using the negative input, the polarity is inverted and can be corrected by setting the ADCx_INVERT bits.

Table 11 shows the settings required for each of the supported analog input channel configurations.

Table 11. Analog Input Channel Configuration Settings

AFEx_PDIFF	AFEx_DIFF	PGAx_EN	CONFIGURATION
0x0	0x0	0x0	Single-Ended (SE) Mode with PGA Disabled (Default)
0x0	0x0	0x1	Single-Ended (SE) Mode with PGA Enabled
0x0	0x1	0x0	Differential (DIFF) Mode with PGA Disabled
0x0	0x1	0x1	Differential (DIFF) Mode with PGA Enabled
0x1	Don't Care	0x0	Pseudo-Differential (PDIFF) Mode with PGA Disabled
0x1	Don't Care	0x1	Pseudo-Differential (PDIFF) Mode with PGA Enabled

Analog Input External Connections

Figure 52 illustrates the typical external connections for a single-ended analog microphone input. The microphone output is AC coupled to the AINx- input pin and the AINx+ pin must be unconnected. For an analog microphone input use case the PGA is typically enabled (PGAx_EN set to 1). For a single-ended line-level input use case replace the AC-coupled microphone output with the line input and disable the PGA.

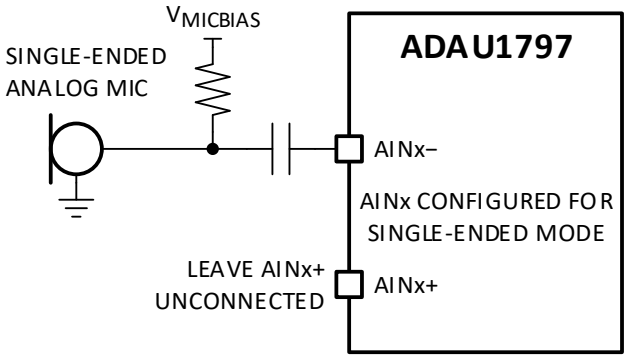

Figure 52. Single-Ended Analog Microphone Input

The device also supports a pseudo-differential mode that provides improved CMRR. *Figure 53* illustrates a polarity inverted use case with a single-ended analog microphone connected to the negative input (AINx-). The unused analog input (AINx+) is AC coupled as close as possible to the ground reference of the single-ended signal source (in this case the analog microphone). For a non-inverted polarity, reverse the positive and negative input connections.

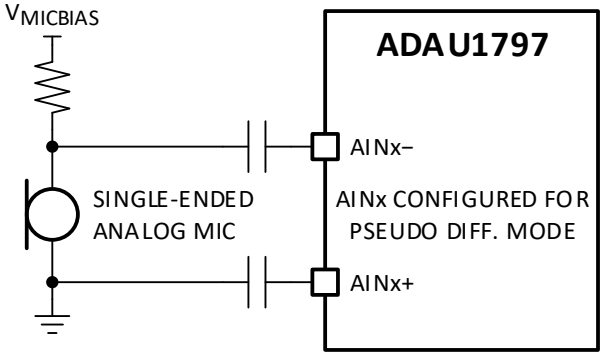

Figure 53. Pseudo-Differential Analog Microphone Input

Figure 54 illustrates a differential analog microphone input use case. The microphone outputs are AC coupled to both the AINx+ and AINx- inputs, and the PGA is typically enabled. For a differential line level input use case, connect the line inputs to the AC coupled input instead, and disable the PGA.

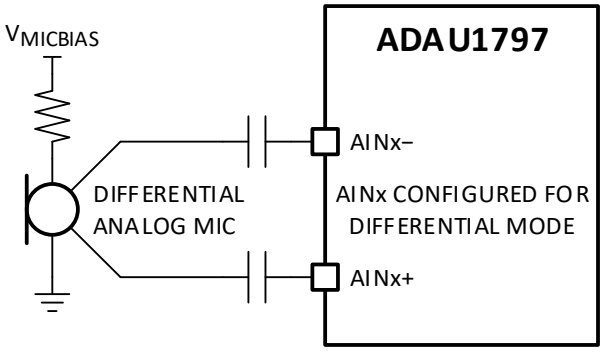

Figure 54. Differential Analog Microphone Input

Analog Input Programmable Gain Amplifier (PGA)

The analog input channels each provide an optional series PGA. When the device is configured for a line-level input, the PGA can be disabled and bypassed (PGAx_EN). When the PGA is disabled, the input impedance is fixed at 21kΩ.

When the device is configured for an analog microphone input with PGA enabled, the gain (for a given analog input channel) is selected with a combination of the PGAx_GAIN and PGAx_SUBGAIN bits. First, the target gain is selected with the PGAx_GAIN bit. The gain is adjustable from a minimum of 0dB (PGAx_GAIN = 0x00) to a maximum of +29.25dB (PGAx_GAIN = 0x27) in 0.75dB steps. Next, set the PGAx_SUBGAIN bits based on the selected analog input mode and the PGA gain level. *Table 12* details the required PGAx_SUBGAIN settings by PGA gain range and input mode.

When using the PGA gain controls, the PGAx_SLEW_DIS bit must always be set high. To avoid audible artifacts the PGA gain level should never be changed while the analog input channel is active. The digital volume control (ADCx_VOL) can be used to implement smooth volume changes while the channel is active. If settings outside of those in *Table 12* are selected, the PGA gain and signal levels may not be as expected, and the audio performance may be degraded.

Table 12. PGAx_SUBGAIN Settings by PGA Gain Range

PGA GAIN RANGE	PGAx_GAIN SETTING	PGAx_SUBGAIN SETTING	
		SE OR PDIFF MODES	DIFF MODE
0dB to +6dB	Don't Care	0b00000	0b00000
+6.75dB to +12dB	0bXXXXX0	0b00000	0b00000
	0bXXXXX1	0b11111	0b00000
+12.75dB to +29.25dB	0bXXXXX0	0b00000	0b00000
	0bXXXXX1	0b11111	0b11111

The PGA can also be configured for coarse attenuation settings (PGAx_MODE). In this mode, the PGA attenuation settings are used instead of the gain settings, and they can be adjusted from 0dB to -9dB in 3dB steps (PGAx_ATTEN).

The internal resistors used for these settings are precisely matched to each other to minimize gain and attenuation errors. However, the exact nominal value of the resistors depends on various conditions in the silicon manufacturing process and can vary by as much as ±20%.

The PGA gain of each analog input channel by default is set individually by the respective PGAx_GAIN bits. To enable synchronous PGA gain changes across all three channels, set the PGA_GAIN_LINK bit to 1. When set, the PGA gain of all three channels matches the channel 0 setting (PGA0_GAIN).

The analog input channel PGA has four power modes: lowest power, low power, normal, and high-performance modes. By default, the PGA operates in normal mode, and the mode is changed with the PGAx_IBIAS bits.

Analog Input Coupling Capacitor Precharge

By default, precharge amplifiers are enabled to quickly charge the large AC coupling capacitors in series with the analog inputs. Precharging these capacitors before enabling the channels can prevent audible signals.

The precharge amplifiers are powered up by default when an analog input ADC channel is enabled (ADCx_EN set to 1) and remain active for the selected precharge time (as set by the ADC_AIN_CHRG_TIME bits). The Precharge time can be disabled or set to one of the available times ranging from 5ms to 400ms. Longer precharge times are needed for larger input coupling capacitors, but this does increase the analog input channel startup times as well.

The internal impedance for the AINx pins is 750Ω during Precharge. However, at startup, the internal impedance is governed by the time constant of the common mode reference voltage (CM pin) because the precharge amplifiers use the CM voltage as a reference.

Analog Input Channel ADCs

Each of the three analog input channels features a high-performance 24-bit, Σ-Δ ADC. Each analog input channel and ADC is enabled with the respective ADCx_EN bit and provides a selectable sample rate, configurable digital filters, and digital volume control.

Analog Input Channel ADC Full-Scale Level

The full-scale analog input (for a 0dBFS output code with 0dB of PGA gain) is nominally $1.08V_{RMS}$ for a differential input. For a single-ended or pseudo-differential input, the full-scale input depends on whether the PGA is enabled. It is $0.64V_{RMS}$ with the PGA disabled and $0.54V_{RMS}$ with it enabled. Input signals that exceed the full-scale level (including PGA gain) cause the channel to clip.

Analog Input Channel ADC Configuration

The output sample rate for the ADC decimation filters can be configured from 8kHz to 768kHz. The sample rate for analog input channels 0 and 1 is linked (ADC01_FS), while the sample rate of channel 2 is individually selected (ADC2_FS). Both high-order (higher delay) and low-order (reduced delay) decimation filters are available. As before the setting for channel 0 and channel 1 is linked (ADC01_DEC_ORDER), while channel 2 is configured separately (ADC2_DEC_ORDER). The output of each channel can be inverted with the ADCx_INVERT bits.

To remove DC offsets, each channel provides a digital high-pass filter (enabled with the ADCx_HPF_EN bits). The HPF cutoff frequency can be configured from 0.25Hz to 241Hz. The setting for channels 0 and 1 is linked (ADC01_HPF_FC), while channel 2 is configured separately (ADC2_HPF_FC).

Compensation filters for the high-frequency roll-off of the ADC decimation filters are provided. These are enabled for channels 0 and 1 with the ADC01_FCOMP bit and for channel 2 with the ADC2_FCOMP bit. Disabling the compensation filters (default) provides the lowest propagation delay, but results in a slight signal attenuation in the passband at higher frequencies.

Analog Input Channel ADC Digital Volume Control

The digital volume of each channel can be set from -71.25dB to +24dB in 0.375dB steps with the ADCx_VOL bits. Each analog input channel can be independently placed into digital mute by setting the corresponding ADCx_MUTE bit high. The channels can also be digitally muted by configuring the digital volume control (ADCx_VOL) to the lowest setting (code 0xFF).

By default, fine-step volume ramping based on zero cross-detection is used for all digital volume changes. Volume change zero cross detection can be disabled with the ADC_VOL_ZC bit, and when disabled each volume step occurs without regard to signal level every 4.5dB/ms. Volume ramping can be bypassed entirely to allow for instantaneous (hard) single-step volume changes by setting the ADC_ HARD_VOL bit.

By default, volume control is independent for each channel. If the ADC_VOL_LINK bit is set, the volume level and any changes for all channels are linked to the channel 0 settings.

When a channel is enabled, it powers up at the volume set by the respective ADCx_VOL bits. When disabled, it powers down immediately without ramping down the volume.

Analog Input Channel ADC Bias and Power Modes

By default, all analog input channel ADCs are configured to operate with a normal bias current in the standard power mode. The ADC bias current can be set to (in order of increasing power consumption and overall performance) extreme power saving, power saving, normal operation, and enhanced performance modes. The overall power mode of all channels is set with the ADC_LP_MODE bit. In low-power mode, a +6dB gain is applied to any channel where the PGA is disabled.

Digital Microphone Input Channels

The device provides two independent digital microphone (DMIC) clock outputs (DMIC_CLK0 and DMIC_CLK1), and five separate digital microphone data input interfaces (DMIC0_1, DMIC2_3, DMIC4_5, DMIC6_7 or DMIC8_9). Each of the DMIC pins shares functionality with a multipurpose pin (MP9 to MP14).

Digital Microphone Input Channel Interface

Each input interface (DMICx_x) can accept two DMIC input channels, allowing for up to ten total digital microphone input channels. Each DMIC data input and signal channel is enabled with the corresponding enable bit (DMICx_EN). The decimation ratio and resulting output sample rate of each digital microphone input channel pair are set by the corresponding DMICxx_FS bit. The output sample rate of each pair can be set from 8kHz to 768kHz.

Each digital microphone data input channel pair (DMICx_x) is mapped to one of the two DMIC clock outputs (DMIC_CLK0 or DMIC_CLK1). The DMIC clock source is mapped independently for each DMIC data input with the respective DMICxx_MAP bit. The DMIC data inputs only support the two DMIC clock outputs and cannot be clocked from another source (such as an external host or audio device).

Each input interface accepts pulse density modulation (PDM) input data. PDM input data is channel interleaved with data for one channel on rising clock edges and the data for the second channel on falling clock edges. The PDM data is mapped directly to the relative pulse code modulation (PCM) data full-scale. For example, data with a 50% PDM density results in a -6dBFS output amplitude (when set to the default DMIC volume setting of 0dB).

By default, the lower channel number of each DMIC input channel pair is clocked on the DMIC clock rising edge while the higher channel is on the falling edge (often denoted as the left channel on the rising edge and the right channel on the falling edge for two microphone systems). The polarity of the active edges for each channel pair can be swapped by setting the corresponding DMICxx_EDGE bit high.

Digital Microphone Input Channel Clocking

Each of the two DMIC clock outputs is independently enabled with the respective DMIC_CLKx_EN bit. The frequency of each clock output is also set individually with the corresponding DMIC_CLKx_RATE bits. The clock output supports a wide range of output frequencies from a maximum of 6.144MHz down to a minimum of 256kHz.

The available DMIC clock frequencies are restricted by the selected output sample rate (set by DMICxx_FS). DMIC clock rates from 256kHz to 4.096MHz (in integer ratios of 2) are only valid with 8/16kHz output sample rates (common voice sample rates). DMIC clock rates from 384kHz to 6.144MHz (also in integer ratios of 2) are only valid with 12/24/48/96/192/384/768kHz output sample rates.

Digital Microphone Input Channel Filters

The DMIC input decimation filter provides both a fourth-order and a fifth-order option. This is selected for each channel pair with the corresponding DMICxx_DEC_ORDER bits. The fourth-order filter provides the lowest propagation delay, while the fifth-order filter improves in-band noise shaping and may be needed to maximize performance with some very high dynamic range digital microphones.

The device provides compensation filters for the high-frequency roll-off of the input decimation filters. These are enabled for each channel pair with the appropriate DMICxx_FCOMP bits. Disabling these filters (default) provides the lowest propagation delay but results in a slight signal attenuation in the passband at higher frequencies.

To remove DC offsets, each DMIC input channel pair also provides a digital high-pass filter (enabled with the corresponding DMICxx_HPF_EN bits). The HPF cutoff frequency can be configured from 0.25Hz to 241Hz. The cutoff frequency setting for each channel pair is selected with the appropriate DMICxx_HPF_FC bits.

The digital microphone input channels and the analog input channel ADCs are completely independent and do not share the same digital decimation filter chains.

Digital Microphone Input Channel Volume Control

The digital volume setting of each digital microphone input channel is independently selected with the corresponding DMICx_VOL bits. The DMIC input channel volume can be set between +24dB and -71.25dB in 0.375dB steps. Each digital microphone input channel can be independently placed into digital mute by setting the corresponding DMICx_MUTE bit high. Alternatively, the channels can also be digitally muted by configuring the digital volume control (DMICx_VOL) to the lowest setting (code 0xFF).

By default, fine-step volume ramping based on zero cross-detection is used for all DMIC channel volume changes. Volume change zero cross-detection can be disabled with the DMIC_VOL_ZC bit. Volume ramping can be bypassed for instantaneous (hard) single-step volume changes by setting the DMIC_ HARD_VOL bit.

By default, volume control is independent for each DMIC input channel. If the DMIC_VOL_LINK bit is set, the volume level and any changes for all channels are linked to the DMIC input channel 0 volume settings.

When a DMIC channel is enabled, it powers up at the volume set by the respective DMICx_VOL bits. When disabled, it powers down immediately without ramping down the volume.

Digital Microphone Input Use Case Example

For a stereo (two DMIC) input use case using a single digital microphone data input interface, the data output of both DMICs should be connected to the same DMIC input pin (DMICx_x). In this case, the same DMIC clock output source (DMIC_CLKx) should be mapped (with DMICxx_MAP) to the selected DMIC input channel pair.

The selected DMIC_CLKx output pin should be connected to the clock input of both DMICs. The clock polarity select pin of one DMIC should be connected for the rising edge and the other for the falling edge. The DMIC physical connection polarity should match the configured clock edge polarity of the DMIC input channel pair (as set with DMICxx_EDGE). *Figure 55* illustrates the typical connections in an example stereo DMIC use case.

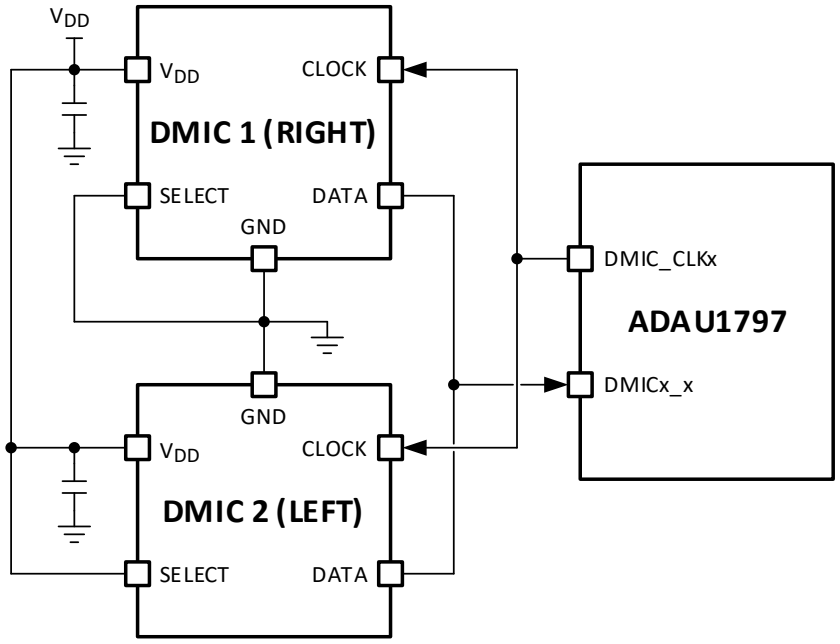

Figure 55. Digital Microphone Stereo Use Case Example

Output Signal Channels

The device provides a single amplifier output channel that includes a high-performance DAC and a low-noise, high-efficiency differential Class-D headphone amplifier.

The device also provides two low latency, high-performance 1-bit PDM output channels suitable for driving an external amplifier or transmitting data to a peripheral device.

Audio data can be routed into any combination of the output signal channels from the outputs of most internal audio subsystem blocks including the serial data ports, serial data input ASRCs, analog input ADC channels, DMIC input interfaces, interpolator filter channels, audio bus map, audio output DMA, FastDSP core, and the HiFi 3z core.

Output Channel DAC

The analog output audio playback channel features a high-performance, low latency, 24-bit resolution Σ-Δ digital-to-analog converter (DAC). The channel is enabled with the PB0_EN bit and provides a selectable input sample rate, configurable digital filters, and digital volume control.

Output Channel DAC Full-Scale Level

The full-scale output (corresponding to a 0dBFS input code with 0dB of digital volume) is nominally $1.15V_{RMS}$ from the DAC to the differential Class-D amplifier output. Output signals that exceed the full-scale input code (with digital volume applied) cause the channel to clip.

The signal level where the channel digitally clips can be reduced (from 0dBFS) to a selected lower level using the DAC_HF_CLIP bits. This setting reduces the full-scale level in increments of 1/256 (single-bit decrement at 8-bit resolution per setting). By default, it is set for no clipping level reduction relative to full scale. The control allows the clipping level to be set from 255/256 (-0.034dBFS) down to 1/256 (-48.165dBFS).

Output Channel DAC Configuration

The analog output channel DAC can accept audio data from other blocks at a range of input sampling rates from 8kHz to 768kHz. The DAC input data sample rate is selected with the DAC_FS bits, and the output of any audio channel routed to the DAC must be configured to the same sample rate. The input data source is selected with the DAC0_ROUTE bits.

The input signal to the channel can be inverted with the DAC_INVERT bit. Both high-order (higher delay) and low-order (reduced delay) input interpolation filters are provided, and this is selected with the DAC_MORE_FILT bit.

To remove DC offsets a digital high-pass filter (HPF) is available and can be enabled with the DAC0_HPF_EN bit. The cutoff frequency can be configured from 0.25Hz to 241Hz and is selected with the DAC_HPF_FC bits.

The device provides optional compensation filters for the high-frequency roll-off of the analog output channel DAC that is enabled with the DAC_FCOMP bit. Disabling the compensation filters (default) provides the lowest propagation delay, but results in a slight signal attenuation in the passband at higher frequencies.

Output Channel DAC Volume Control

The digital volume can be set from -71.25dB to +24dB in 0.375dB steps with the DAC0_VOL bits. The channel can be placed into digital mute by setting the DAC0_MUTE bit high. The channel can also be muted by setting the digital volume control (DAC0_VOL) to the lowest setting (code 0xFF).

By default, fine-step volume ramping based on zero cross-detection is used for all digital volume changes. Volume change zero cross detection can be disabled with the DAC_VOL_ZC bit, and when disabled each volume step occurs without regard to signal level at 4.5dB/ms. Volume ramping can be bypassed entirely to allow for instantaneous (hard) single-step volume changes by setting the DAC_ HARD_VOL bit.

When the channel is enabled, it powers up at the volume set by the DAC0_VOL bits. When disabled, it powers down immediately without ramping down the volume.

Output Channel DAC Power Modes

Three power mode options trade off varying degrees of performance with power consumption.

The DAC power mode bits (DAC_PWR_MODE) can be set to normal, low-power saving, and high-power saving modes. The low-power modes offer a similar level of performance at reduced power, however, the lowest-power mode increases channel latency by about 1µs.

The DAC can also be placed into a high-performance mode by setting the DAC_PERF_MODE bit high. In this mode, THD+N is improved at high signal amplitudes at the cost of slightly increased power consumption.

The DAC bias current can be adjusted with the DAC_IBIAS bits and can be set to normal operation, power saving, or enhanced-performance modes. The lower power settings result in slightly increased distortion.

Output Channel Class-D Amplifier

The device features a Sigma-Delta Pulse Density Modulation differential Class-D amplifier with very low output noise and distortion. The amplifier has very low quiescent power, enabling very high active power efficiency over the full output power range.

The differential headphone amplifier output (HPOUTP and HPOUTN) can drive a minimum load of 6Ω. The common mode stable output does not typically require any output filter to directly drive the speaker load, saving board space and reducing component count. In addition, the modulation scheme reduces the amplitude of spectral components at high frequencies, reducing EMI emissions that can otherwise be radiated by speakers and cable traces.

Class-D Amplifier Operation

The Class-D amplifier is automatically enabled when the analog output channel is enabled with the PB0_EN bit. The amplifier does not provide any direct analog gain or an analog mute control. The overall volume (or gain) of the analog output channel should be set with the digital volume control (DAC0_VOL) and if output mute is needed the digital mute control must be used (DAC0_MUTE).

The differential load between HPOUTP and HPOUTN must exceed the specified minimum load resistance and inductance to ensure that power and performance specifications are met. To optimize the amplifier output power stage performance, the output resistive load bits (HP_RLOAD) should be set to the closest matching value to the attached load resistance.

The amplifier defaults to high-performance mode but can be placed in low-power mode by setting HP_LPM = 1. This reduces the Class-D modulator power consumption at the expense of increased THD+N and output noise.

Class-D Amplifier Pop-and-Click Suppression

The analog output channel and amplifier architecture provides advanced pop-and-click suppression that minimizes any potentially audible transients during channel power-up (activation) and power-down (deactivation). To minimize pop-and-click, the proper sequencing should be followed.

Before the amplifier output power up, the audio clocks should be active and stable, and the input data should either be silent (zero code) or the channel should be muted (DAC0_MUTE). Once the power-up sequence is complete, the channel can be unmuted and playback can begin. If the internal digital volume control is used for soft ramping, then the channel should be unmuted once audio data is present. Alternatively, if the host is ramping up the audio data, then the channel can be unmuted first before sending ramped audio data.

Before power-down, the input data to the channel should be muted (for soft ramp-down) or ramped down by the host to silent input data. The audio clocks must not be disabled until the power-down sequence is completed. Changing any output channel settings (outside of volume and mute) while playback is active may result in audible transients. These changes should only occur when the channel is either muted or powered down (disabled).

Class-D Amplifier EMI Management

The amplifier uses a proprietary common-mode stable output switching, modulation, and spread-spectrum technology to minimize EMI emissions. This results in lower emission levels than other filterless Class-D topologies. The distance from the amplifier outputs to the speaker load impacts the radiated emissions from the board and/or cables. For some applications with longer trace or cable lengths, a small ferrite bead filter may be required for EMC compliance. Typically for trace or cable lengths less than 4" no extra filter components are needed.

For additional EMI emissions reduction, the device provides a Class-D output slew rate control (HP_EDGE) that when set, places the outputs into low EMI mode. This mode significantly reduces the output radiated emissions (particularly above 30MHz) at the expense of reduced output power efficiency.

If an output filter is required (or for extended trace lengths), amplifier output feedback connections (FB_HPOUTP and FB_HPOUTN) should be connected post-filter to correct for any non-linearities of the filtering components. If no filter is used, they can either be connected directly to the appropriate headphone output pins (HPOUTP and HPOUTN) or to a point in the trace or wiring that is closer to the speaker load.

Class-D Amplifier Fault Protection

The analog output headphone amplifier includes fault detection, protection, and status reporting for amplifier over current (short circuit) and over-temperature faults. Headphone amplifier output over current protection is enabled by default but can be disabled by setting HP_OCP_EN = 0 (not recommended).

Output over current protection can detect an output short to GND or supply (which reports the status and disables the analog output channel amplifier). However, if HPOUTP and HPOUTN are shorted together, there can be instances where the overcurrent protection is not triggered resulting in significant HPVDD current. If it is necessary to protect against an output short, then system-level protection is recommended.

Both over-current and over-temperature faults are by default set to auto-recovery mode. In auto mode, the headphone amplifier automatically attempts to recover and re-enable after being shut down during a fault event.

To place over-current and over-temperature protection into the manual-recovery mode, set the HP_ARCV_SC and HP_ARCV_OT bits high (respectively). In manual mode, after a fault occurs and is detected (with the status bits and/or interrupts), the amplifier output is disabled and requires driver action to re-enable it. To manually attempt to restart the amplifier, the system software can either toggle the playback enable bit (set PB0_EN to disabled, then back to enabled) or switch the device into auto mode (until recovery occurs, then it can be placed back into manual mode).

PDM Output Channels

The device provides two low latency, high-performance 1-bit PDM output channels suitable for driving an external amplifier or data to a peripheral device. The outputs are enabled with the corresponding PDMx_EN bits.

The two PDM output channels share a single output clock and data interface. PDM channel 0 data is clocked on rising edges, and PDM channel 1 data is clocked on falling edges. The shared PDM clock and data outputs can each be individually routed to one or more multipurpose pins (MP0 to MP31) with the respective MPx_MODE bits.

PDM Output Channel Full-Scale Level

Full-scale input data (corresponding to a 0dBFS input code with 0dB of digital volume) results in full-scale output PDM data. The performance of the PDM modulator is degraded for PDM output amplitudes greater than -7.5dBFS.

PDM Output Channel Configuration

The PDM output channels can accept audio data from other blocks at a range of input sampling rates from 12kHz to 768kHz. The input data sample rate for both channels is selected with the PDM_FS bits, and the internal audio data routed to each channel must be configured to the same sample rate. The input data source for each PDM output channel is selected with the respective PDMx_ROUTE bits.

The PDM output channel modulators can operate at sample rates of 3.072MHz, 6.144MHz, or 12.2888MHz. The PDM data sample rate is always equal to the PDM output clock frequency and is selected with the PDM_RATE bits.

The input signal to the PDM channels can be inverted with the respective PDMx_INVERT bits. Both high-order (higher delay) and low-order (reduced delay) input interpolation filters are provided, and this is selected with the PDM_MORE_FILT bit. To remove DC offsets digital high-pass filters are available for each PDM channel and can be enabled with the respective PDMx_HPF_EN bits. The cutoff frequency can be set from 0.25Hz to 241Hz and is selected with the DAC_HPF_FC bits.

Optional compensation filters for the high-frequency roll-off of the channels are enabled with the PDM_FCOMP bit. Disabling the compensation filters (default) provides the lowest propagation delay but results in a slight signal attenuation in the passband at higher frequencies.

PDM Output Channel Digital Volume Control

The digital volume of each channel can be set individually from -71.25dB to +24dB in 0.375dB steps with the appropriate PDMx_VOL bits. Each channel can be placed into digital mute by setting the respective PDM0_MUTE bit high. They can also be muted by setting the digital volume control (PDMx_VOL) to the lowest setting (code 0xFF).

By default, fine-step volume ramping based on zero cross-detection is used for all digital volume changes. Volume change zero cross detection can be disabled with the PDM_VOL_ZC bit, and when disabled each volume step occurs without regard to signal level at 4.5dB/ms. Volume ramping can be bypassed entirely to allow for instantaneous (hard) single-step volume changes by setting the PDM_ HARD_VOL bit.

By default, volume control is independent for each PDM output channel. If the PDM_VOL_LINK bit is set, the digital volume level and any changes for both channels are linked to the PDM output channel 0 volume settings (PDM0_VOL). When a channel is enabled, it powers up at the volume set by the respective PDMx_VOL bits. When disabled, it powers down immediately without ramping down the volume.

Interpolation and Decimation Filters

In addition to the decimation and interpolation filters that are attached to specific channels, the device provides an additional standalone set of eight decimation filters and eight interpolation filters. These filters can accept output data routed from and provide input data to most internal audio subsystem blocks.

The eight decimation filter channels convert higher sample rate input data (fast) to lower sample rate output data (slow) and can each be enabled individually with the FDECx_EN bits. The input data source is selected individually for each channel and is routed with the corresponding FDECx_ROUTE bits. The input and output sample rate settings for the eight decimation channels are grouped in pairs (channel pairs 0/1, 2/3, 4/5, and 7/8). The input and output sample rate for each pair is set with the respective FDECxx_IN_FS bits (input) and FDECxx_OUT_FS bits (output). The selected input sample rate must be higher than the output sample rate.

The eight interpolation filter channels convert lower sample rate input data (slow) to higher sample rate output data (fast) and are each enabled individually with the FINTx_EN bits. The input data source is selected individually for each channel and is routed with the corresponding FINTx_ROUTE bits. The input and output sample rate settings for the eight interpolation channels are grouped in pairs (channel pairs 0/1, 2/3, 4/5, and 7/8). The input sample rate for each pair is set with the respective FINTxx_IN_FS bits and the output sample rate is set with the respective FINTxx_OUT_FS bits. The selected input sample rate must be higher than the output sample rate.

The interpolation and decimation filter channels support input and output sample rate settings ranging from 8kHz to 768kHz, however, only input-to-output sample rate ratios that do not have a fractional component are supported. For example, a decimation filter with an input sample rate of 16kHz and an output sample rate of 24kHz is not supported since the ratio (24kHz/16kHz) has a fractional component.

Asynchronous Sample Rate Converters

The device provides two full-duplex (input and output) asynchronous sample rate converters (ASRCs) that can each be paired with either of the two serial audio data ports. Each ASRC (ASRC 0 and ASRC 1) provides four digital input and four digital output channels (for a total of eight digital input and eight digital output ASRC channels).

Each ASRC input and output channel is individually enabled. For ASRC 0 the output channels are enabled with the ASRCO0_x_EN bits and the input channels are enabled with the ASRCI1_x_EN bits. For ASRC 1 the output channels are enabled with the ASRCO1_x_EN bits and the input channels are enabled with the ASRCI1_x_EN bits.

ASRC Configuration

The ASRCs convert input and output audio data to or from the sample rate of the paired serial audio data port to or from the selected internal synchronous audio data sample rate. The serial port paired with each of the two ASRCs is selected with the ASRCOx_SAI_SEL and ASRCIx_SOURCE bits for the output and input channels respectively.

When paired with the ASRCs, the serial audio data ports can support external asynchronous input and output audio data at sample rates from 7kHz to 224kHz. All intermediate frequencies and clock ratios are supported. For voice bandwidth external sample rates (typically nominal sample rates from 8kHz to 32kHz), additional low pass filtering can be enabled with the ASRCIx_VFILT (for input channels) and ASRCOx_VFILT (for output channels) bits.

For each ASRC, the sample rate for the input and output channels is individually selected from a discrete set of values ranging from 8kHz to 192khz. The internal sample rate accepted (ASRCOx input rate) by the four output channels is linked and is set for the four output channels with the ASRCOx_IN_FS bits. The sample rate provided (ASRCIx output rate) by the four input channels is also linked and is set with the ASRCIx_OUT_FS bits.

For both ASRCs, the four input channels and four output channels will automatically mute the converted output data to zero code if they have not locked or lose lock. The lock-on state of the input and output channels for each ASRC can be monitored with the corresponding ASRCIx_LOCK (for input channels) and ASRCOx_LOCK (for output channels) read-only status bits. The unlocked-to-locked or locked-to-unlocked transitions for each ASRC can be used as interrupt sources for the two interrupt controllers.

ASRC Signal Routing

Audio data can be routed to each ASRC input channel from any of the input channels of the paired serial audio data port (as set by the ASRCIx_COURSE bits). The source audio data for each ASRC input channel is selected with the respective ASRCIx_x_ROUTE bits.

Similarly, audio data can be routed into the ASRC output channels from the outputs of most internal audio subsystem blocks including the analog input ADC channels, DMIC input channels, decimator filter channels, audio bus map, audio output DMA, Fast DSP core, and the HiFi 3z core. The source audio data for the ASRC output channels is selected with the respective ASRCOx_x_ROUTE bits. The internal audio data routed to each ASRC output channel must be configured to the same sample rate as the ASRC output channel.

ASRC Power Modes

By default, the input and output channels of both ASRCs operate in high-performance mode. Two lower power modes with slightly reduced performance are provided. Generally, if the ASRC input or output data is from or to the ADC or DAC the low-power modes do not degrade the signal below the performance of the converters. For ASRC input channels these modes are enabled with the ASRCIx_LPM (low power) and ASRCIx_LPM_II (lowest power) bits. For output channels, they are enabled with the ASRCOx_LPM (low power) and ASRCOx_LPM_II (lowest power) bits.

Table 13. Input ASRC Power and Performance Options for 44.1kHz to 48kHz Conversion

MODE	THD+N at 1kHz	THD+N at 20kHz	DNR A-WEIGHTED	POWER PER CHANNEL (mW)
Default	-130dB	-120dB	130dB	0.174
ASRCI_LPM = 1	-120dB	-110dB	130dB	0.130
ASRCI_LPM_II = 1	-115dB	-90dB	130dB	0.108

Table 14. Output ASRC Power and Performance Options for 48kHz to 44.1kHz Conversion

MODE	THD+N at 1kHz	THD+N at 20kHz	DNR AW	POWER PER CHANNEL (mW)
Default	-130dB	-120dB	130dB	0.452
ASRCO_LPM = 1	-120dB	-110dB	130dB	0.289
ASRCI_LPM_II = 1	-115dB	-90dB	130dB	0.205

Boot Loader

The device employs a boot loader for loading an app pack into system memory. This boot loader is used both in self-boot mode and host-boot mode.

To access the HiFi 3z memory, FDSP memory, or Main Map registers, the boot loader needs to be enabled. This is true even if an app pack is not present in the system memory.

Boot Loader in Host-Boot Mode

In host-boot mode (SELFBOOT pin = GND), an application pack can be written externally via I^2C or SPI into the L2 memory. Once the memory is configured, the PROC_EN bit must be asserted. This enables the boot loader, which then in turn enables the HiFi 3z core.

If the application pack is not successfully loaded or the L2 memory is left blank, then the PROC_EN bit must still be asserted to enable the boot loader. Once the boot loader has been enabled, access to the memories and main control registers is open.

When operating in host-boot mode, make sure to perform the following steps before writing to the L2 memory:

- Assert DVDD_EN
- Set all clock configuration registers
- Update and enable the PLL
- Assert MASTER_BLOCK_EN

Once these steps are completed, write the app pack to the L2 memory and then assert PROC_EN. If nothing is written to the L2 memory, PROC_EN still needs to be asserted to gain access to memories and main control registers.

Boot Loader in Self-Boot Mode

In self-boot mode (SELFBOOT pin = V_{IOVDD}), an application pack can be loaded via external QSPI Flash.

If the self-boot is unsuccessful or the external flash is empty, the boot loader is enabled during the attempted self-boot. As a result, access to the memories and main control registers is open.

In self-boot mode, the internal DVDD defaults to a value of 1.1V. The boot loader needs 1.1V to attempt the boot and enable the HiFi 3z core. Once the boot loader has finished the attempted boot, then DVDD can be switched to 0.9V. Likewise, if using an external DVDD in self-boot mode, the DVDD voltage provided must be 1.1V.

In self-boot mode, the default clock configuration values assume an MCLK frequency of 24.576MHz. Hence the MCLK must be 24.576MHz during the attempted boot. Once the boot loader is finished, the MCLK frequency can be changed and the clock configuration register can be set with new values.

FastDSP Core

The device features a proprietary FastDSP core optimized for low-latency audio processing use cases such as active noise cancellation and ambient transparency. The FastDSP is designed to be intuitively configured with the *SigmaStudio+* software. The core is controlled with a 27-bit program word and supports a maximum of 128 instructions per frame. The configured clock speed, DVDD voltage level, and frame rate source limit the number of instructions that can be run in each frame (*Table 15*).

Each FastDSP core instruction is configured independently, and the supported functions include biquad filters, limiters, expanders, multipliers, bitwise operations, clippers, volume controls, and weighted mixing. The Fast DSP instructions can accept input source data channels (external to the core) from the ADC inputs, DMIC inputs, Interpolators, ADMA, AMAP, HiFi 3z TIE, serial audio data ports, and ASRC inputs.

The FastDSP core provides sixteen output data channels. While the FastDSP core can maintain up to 24dBFS of headroom internally, it clips symmetrically to 0dBFS at each output channel. The FastDSP provides output clip detectors on each output channel that can be read as status bits or used as an IRQ source. By default, there is no gain adjustment between any instruction or block.

FastDSP Power and Run Control

Setting the FDSP_EN high enables the FastDSP and allows access to the core memories. All program, parameter, and data memories for the FastDSP can be read or written from any control interface or the HiFi 3z DSP when POWER_EN = 1, FDSP_EN = 1, and the PLL is locked (if in use). The FastDSP starts processing data when it is both enabled (FDSP_EN = 1) and set to run (FDSP_RUN = 1).

FastDSP Clock Speed and Frame Rate

The FastDSP core clock speed supports operation at either 24.576MHz or 49.152MHz, and this is selected with the FDSP_SPEED control bit. The 24.576MHz clock speed setting can be supported with a nominal DVDD supply voltage level of either 0.9V or 1.1V, however, the 49.152MHz setting only operates with a DVDD supply level of 1.1V. The maximum number of FastDSP instructions per frame based on the selected configuration is shown in *Table 15*.

Table 15. Maximum FastDSP Instructions per Frame

FastDSP CLOCK SPEED	DVDD LEVEL	FastDSP SOURCE FRAME RATE		
		≤ 192kHz	384kHz	768kHz
24.576MHz	0.9V or 1.1V	128 Instructions	64 Instructions	32 Instructions
49.152MHz	1.1V Only	128 Instructions	128 Instructions	64 Instructions

The source frame rate of the FastDSP is selected with the FDSP_RATE_SOURCE bits, and this determines when the program counter starts counting each frame at 0. The frame rate must be set equal to the audio data sample rate of the fastest input source to any instruction. Supported frame rate sources include ADC channels, DMIC channels, serial audio data ports, input ASRC channels, and interpolator channels. If the fixed source setting is selected, the FastDSP frame rate can be set independently of any source. In this case, the frame rate is set relative to the clock speed using the FDSP_RATE_DIV bits.

FastDSP Input Sources

Any instruction can use any of the following as an input source: any digital microphone or ADC input channel, any serial port input channel, any ASRCI input channel, any interpolation filter channel, any HiFi 3z output channel, any audio output DMA channel, or any other FastDSP instruction output (routed through a data or accumulator register).

FastDSP Data Channels and Memory

The FastDSP core internal data channels are 28-bits (5.23 format), and provide sufficient headroom for up to 24dBFS of signal swing before clipping occurs. All input sources to and output channels from the FastDSP are limited to 24 bits (1.23 format). The FastDSP provides 16 output channels each of which is truncated to 24 bits. If any output channel data exceeds full scale, it clips symmetrically to 0dBFS. The FastDSP provides output clip detectors on each channel that can be read as status bits or used as an IRQ source.

FastDSP Supported Instructions

The function of each instruction is individually configured, and a complete list of supported instructions is provided in the *SigmaStudio+* software. The available instructions include the following:

- Single precision (27-bit fractional precision) biquad/second-order filters
- Double precision (54-bit fractional precision) biquad/second-order filters
- Lower precision (19-bit fractional precision) biquad/second-order filters
- Limiter with/without external detector loop or side chain input
- Expander with/without external detector loop or side chain input
- Ramped volume slider
- Mute function
- Sample-based time delay function
- Two input multiply function
- Linear gain function
- Two to four input weighted mixer
- Two to four input addition
- Symmetrical clipper
- Absolute value function
- Two input min and max value functions
- Bit shift function
- Bitwise and, or, xor, and invert functions
- Memory read or write
- Input to output equivalence function
- T connection for signal routing in *SigmaStudio+*

FastDSP Conditional Execution

Each instruction can be set to either always execute, or to only execute conditionally based on a specific flag result or upon certain state conditions. When an instruction does not execute (based on a condition), it can be set to either do nothing or pass its input to its output. Each instruction can generate flags for conditional execution that are based on the output of that instruction. Instruction output flag set conditions include the following:

- Output equals zero
- Output is not equal to zero
- Output is greater than zero
- Output is less than zero
- Output is greater than or equal to zero
- Output is less than or equal to zero
- Accumulator overflow

Each instruction can also conditionally execute based on the current value of certain state conditions. Instruction conditional execution states include the following:

The logic state of any multipurpose pin (MP0 to MP31) when used as a GPIO input (for a given MPxx input when configured with the corresponding MPxx_MODE bits set to 0x01). In addition, the state of any multipurpose pin configured as a GPIO output (MPxx_MODE bits set to 0x02) can be set in with the GPIOxx_OUT bits.

The state of the FastDSP generic conditional execution register bits (FDSP_REG_COND0 to FDSP_REG_COND7). Each of these bits is read/write, and to control FastDSP execution they can be configured with the control interfaces (I^2C or SPI) or directly by the HiFi 3z DSP.

The Modulo N counter equals zero. The Modulo N counter increments once for every FastDSP frame. The counter is reset to 0 after the number of frames completed is equal to the setting of the FDSP_MOD_N bit. Conditional instructions can then execute every N frame (as set by the FDSP_MOD_N bit), and this results in these instructions running at a lower rate than the configured FastDSP core frame rate.

FastDSP Filter Precision

FastDSP core instructions can be configured as biquad filters and second-order filters with a selectable level of fractional precision. These filters can be either single precision (27-bit), double precision (54-bit), or lower precision (19-bit), where using reduced fractional precision results in lower power consumption. However, care must be taken to ensure that filters have enough precision to maintain stability and create the desired filter response.

FastDSP Parameters

Each FastDSP instruction (up to a maximum of 128 instructions) has 5 associated parameters (for a maximum total of 640 parameters in a single bank). Each parameter is stored in memory as a 32-bit number, and the format of a parameter depends upon the associated instruction. Parameters contain instruction configuration information such as filter coefficients, limiter threshold and ballistic settings, and volume control settings. For example, the 5 parameters for a biquad filter instruction contain the biquad filter coefficients (B0, B1, B2, A1, A2) in a 5.27 format.

Reference the *SigmaStudio+* software for the full list of supported instructions and the associated configuration settings and parameters for each. When instantiating FastDSP instructions with the *SigmaStudio+* software, the assembler automatically maps the instruction configuration settings to the associated parameters in memory. Individual sets of 5 parameters in the parameter memory space are sequentially assigned to instructions in the order in which they are instantiated in the FastDSP code.

FastDSP Parameter Bank Switching

The FastDSP provides three separate banks of parameters designated as parameter bank A, bank B, and bank C. Each FastDSP parameter bank supports a maximum total of 128 instructions (each of which has 5 parameters for a total of 640 parameters per bank). The three banks each support the same single set of instructions but allow for instruction parameters such as filter coefficients, settings, and variables to easily be switched between for different use cases and processing scenarios.

At any given time, the FastDSP actively uses only one of the three-parameter banks. The FastDSP can be switched between Bank A, Bank B, and Bank C on the fly while the core is running. The active parameter bank for the FastDSP is selected with the FDSP_BANK_SEL bits. Instruction parameters in the inactive banks can be updated at any time, however, audible glitches may occur if parameters in the active bank are directly updated while the core is running. Parameters in the active bank can only be safely updated with a FastDSP safeload.

The bank change transition method is determined by the FDSP_RAMP_MODE bit setting. When the active parameter bank is changed (with the FDSP_BANK_SEL bits), the parameter values used for instruction processing can either be instantly changed at the start of the next frame or they can be ramped via linear interpolation between the previously selected bank and the new selected bank.

When the linear ramp mode is selected, the rate at which the ramp between the two banks occurs is selectable via the FDSP_RAMP_RATE bits. Only the parameters associated with the three biquad filter instructions are ramped. All other parameters associated with other instructions change at the beginning of the frame where the bank switch occurs and ramping begins, and parameters in banks that are actively ramping do not change during a bank switch.

The FDSP_ZERO_STATE bit can be set to clear the state of the FASTDSP memory during a bank switch. During a bank switch, this prevents the new filter settings (of the new bank) from using old data (from the previous bank) that can be recirculating in the filters and may prevent potential filter instability or audible glitches.

It is possible to stop the linear ramp of biquad filter parameters between the values in the previous and the values in the current target bank. The 6-bit FDSP_LAMBDA setting selects the point along the linear interpolation curve between the two banks at which the bank switch ramp pauses (with 0 being the beginning and 63 being the end of the ramp). To complete a bank switch without pausing set a value of 63 (default), and to pause midway set a value of 31.

The lambda value can be updated actively with the control interfaces, however once a ramped bank switch is in progress it can only be increased. The progress of the current active ramp (from 0 to 63) can be read at any time with the FDSP_CURRENT_LAMBDA bits. When this value reaches 63, the bank switch is complete, and all current parameters in use match those of the target bank. Parameters in the two banks being ramped between cannot be modified while a ramped bank switch is occurring.

An interrupt for either interrupt controller can be triggered via the IRQx_PRAMP interrupt source bits. This triggers on the first frame when a ramped bank switch is active and FDSP_CURRENT_LAMBDA equals FDSP_LAMBDA.

FastDSP Parameter Bank Copying

The instruction parameters of any given bank (source) can be copied to any other bank (target) with a single-bit write command. There are six FDSP_COPY_xy bits (where x is the source bank and y is the target bank), one for each of the six possible bank copy operations. Writing a 1 to one of these bits initiates a bank copy. Once initiated, the bank copy operation waits until the start of the next FastDSP frame and then copies the parameter content of the source bank to the destination bank while the instructions are executed. The bank copy completes at the start of the subsequent frame and takes at most two frames to complete from the initiation. Copying to the active bank (copy target) is not permitted and results in no action being taken.

FastDSP Parameter Memory Access

If the FastDSP core is enabled but not running, then reads from any parameter memory bank through the I^2C interface, SPI interface, or HiFi 3z DSP are unrestricted. However, when the core is enabled and running, only reads from the inactive parameter banks are unrestricted. While the FastDSP core is running, if multiple sources try to read the same memory location on the same cycle, the HiFi 3z DSP has priority over the I^2C and SPI interfaces, and the read attempt from the I^2C interface or the SPI interface returns all 0s. Direct reads from the memory of the active bank by any source (I^2C interface, SPI interface, HiFI 3z DSP, or mREAD instructions) are not allowed and return 0s.

Similarly, writes to all parameter banks are possible when the FastDSP core is enabled but not running, and are still permitted to inactive banks while the FastDSP is running. While the core is running, if multiple sources try to write to the same location on the same cycle, the HiFi 3z DSP has priority and the other writes do not occur.

Table 16. Memory Addressing for FastDSP Core

MEMORY	MEMORY SIZE	WORD SIZE	BASE ADDRESS (Hex)
Bank A Parameter 0	128	32	0xF0020000
Bank A Parameter 1	128	32	0xF0020080
Bank A Parameter 2	128	32	0xF0020100
Bank A Parameter 3	128	32	0xF0020180
Bank A Parameter 4	128	32	0xF0020200
Bank B Parameter 0	128	32	0xF0020280
Bank B Parameter 1	128	32	0xF0020300
Bank B Parameter 2	128	32	0xF0020380
Bank B Parameter 3	128	32	0xF0020400
Bank B Parameter 4	128	32	0xF0020480
Bank C Parameter 0	128	32	0xF0020500
Bank C Parameter 1	128	32	0xF0020580
Bank C Parameter 2	128	32	0xF0020600
Bank C Parameter 3	128	32	0xF0020680
Bank C Parameter 4	128	32	0xF0020700
State 0 (A1 High)	128	32	0xF0030000
State 1 (A2 High)	128	32	0xF0030080
State 2 (A1 Low)	128	32	0xF0030100
State 3 (A2 Low)	128	32	0xF0030180
Program	128	32	0xF0040000

FastDSP Parameter Safeload

The parameter safe load mechanism allows the parameter memory for a single instruction in the active bank to be updated while the FastDSP is running. First, Set the target instruction number with the FDSP_SL_ADDR bits (numbered in order of instantiation). Then set the new parameter values with the FDSP_SL_Px bits where x is numbered from 0 through 4 for the 5 parameters associated with the selected instruction number. Finally, write a 1 to the FDSP_SL_UPDATE bit to initiate the safeload. All parameters for the selected instruction are updated at the same time at the beginning of the next frame.

Utilize the *SigmaStudio+* software interface and FastDSP program export to identify the correct instruction number and parameter values for a given desired use case. For example, an export can be performed for instruction with the normal settings and updated settings to identify both sets of parameter values needed for a direct parameter safeload operation. Parameter safeload should be treated as an extra FastDSP instruction, and to enable using it at least one instruction must be unused (out of the maximum support instructions for a given use case, see *Table 15*).

There is a second FastDSP safeload interface that is mapped to the data memory space of the HiFi 3z DSP, which allows the HiFi 3z DSP to have word-addressable access.

HiFi 3z DSP Core

Overview

The HiFi 4 core features a 32-bit, audio DSP engine optimized for audio and voice processing and other demanding DSP functions. The HiFi 3z core combines two 32-bit × 32-bit fixed point MACs or four 24-bit × 24-bit fixed point MACs or four 32-bit × 16-bit fixed point MACs or up to eight 16-bit × 16-bit fixed point MACs, a single 32-bit IEEE floating-point multiplier, and support for dual 64-bit load/store per cycle. The processor provides on-chip debug support with control of the software state of the processor through an IEEE 1149.1 test access port, also known as JTAG. The device is software-compatible with a comprehensive ecosystem of HiFi architecture-optimized audio and voice codecs and audio enhancement software packages.

Programming

On power-up, the device must be configured with a clocking scheme and then loaded with register settings. After the codec signal path is set up, the DSP core can be programmed. The device can be programmed using the *SigmaStudio+* graphic tool provided by Analog Devices or in C++. No knowledge of writing line-level DSP code is required for graphical programming. More information about *SigmaStudio+* is available at *www.analog.com/SigmaStudio+*.

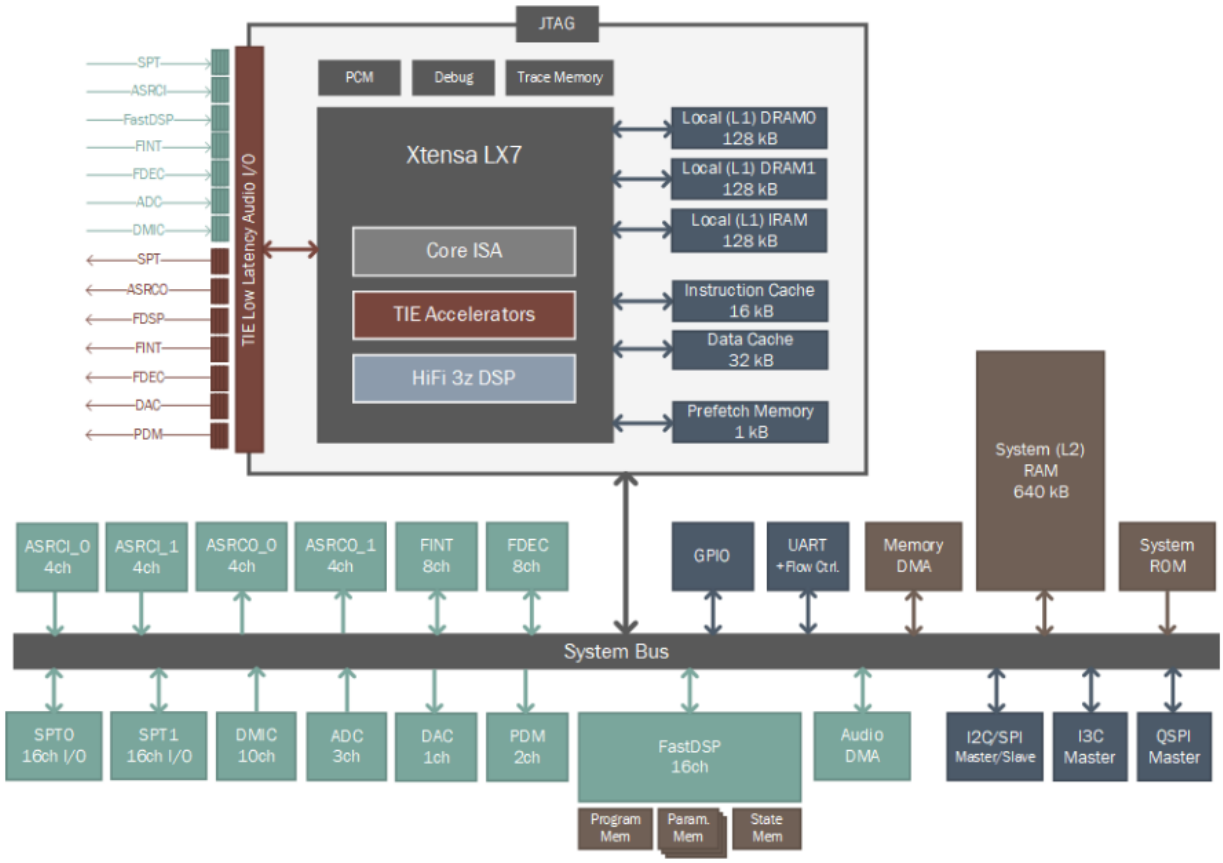

Figure 56. LX7/HiFi 3z IO, Bus, and Memory Structure

Clock Speed Control

By default, HIFI_SPEED is set to 0, the HiFi 3z DSP receives a 49.152MHz clock, and a DVDD nominally of 0.9V is applied. If the PLL is used and is set to 1 then a nominal DVDD of 1.1V must be used, the HiFi 3z DSP core receives a 196.608MHz clock and can run four times as many instructions. If the extra processing power is not needed, then operating with HIFI_SPEED = 0 reduces overall power consumption.

HIFI_SPEED can be changed during operation, but if this is done then the PLL must always be operating with a 196.608 MHz output. PLL_BYPASS cannot be selected if HIFI_SPEED is changed while the part is operating.

Interrupts

The HiFi core has several internal core interrupts.

- Three timer interrupts
- Bus write error
- Profiling interrupt
- 5 Software interrupts (1 at each priority level)
- Non-Maskable Interrupt (NMI)

In addition to the core interrupts, there are three additional external interrupt types comprising a total of 20 external interrupts.

Nine system interrupts aggregate interrupts from various system events and peripherals. The system interrupt controller can individually mask each interrupt source to every system interrupt. System interrupts can interrupt the core and/or also signal an interrupt on an MP pin. MP interrupt controls are described in the *Multipurpose Pins as Interrupt Outputs* section. The interrupt sources for system interrupts include.

- QSPI port and its Rx and Tx DDEs
- UART Rx and Tx DDEs
- Audio DMA engine, which has separate internal interrupt sources.
- FastDSP memory read and write access errors
- Memory copy read and write DMA/DDEs
- Watchdog counter
- Power-up complete signal (only useful for pin interrupts)
- PLL locking and unlocking
- Headphone amplifier protections and faults
- AVDD undervoltage detection
- ASRCs locking and unlocking
- ADC, DAC, and FastDSP output clipping detection
- Generic HiFi interrupts that the core can set through registers to aggregate interrupts from the core and other sources to an MP pin interrupt.

Which interrupt source triggered each system interrupt can be determined by the IRQx_STATUS registers in the READ_ONLY register map. All sources of each interrupt are cleared via a write of 1 to the IRQx_CLEAR bits. The interrupt status bits are sticky, such that if an interrupt source becomes true, the status reads 1 until a clear occurs, even if that interrupt source is no longer true.

Pin interrupts can either be edge or level-sensitive, there are three of each type. Each pin interrupt can map any single pin to its source. The routing of the pins to the interrupt source is set via the HIFI_EDGE_IRQx_SRC and HIFI_LEVEL_IRQx_SRC registers. Each of these interrupt pin sources can also be inverted if desired.

Audio interrupts are generated by the various audio source peripherals when a new sample is ready. Each audio interrupt can map to any single audio source via its HIFI_AUD_IRQx_SRC setting. Additionally, these can be set to block mode by setting HIFI_AUD_IRQx_TYPE to 1 to interrupt after a number of samples determined by HIFI_AUD_IRQx_CNTR are received. If HIFI_AUD_IRQx_TYPE is set to 0 and counter is selected as the source then an interrupt occurs every number of 24.576MHz (40.69ns) clock cycles determined by the HIFI_AUD_IRQx_CNTR, effectively acting as a timer interrupt.

Audio DMA

The three audio DMA blocks support moving up to 8-channels of audio into and out of the processor memory space. The audio data is stored and retrieved as interleaved data.

The DMA block supports a read and a write circular buffer pointer. The circular buffer pointer consists of an index-, start-, and end-address. The index address always points to the start of the current sample. The start- and end-address are the beginning and end of the circular buffer. For non-circular buffer mode, the end address is set to 0 [default]. All addresses are byte addresses.

Each DMA supports up to 8-channels. The channels are configured independently, such that we have up to 8 input channels and up to 8 output channels. The channel enable configuration is a simple 7-bit value. Which audio peripheral source channels are routed into the Audio DMA are selectable via routing registers. Outputs from the Audio DMA to Audio sink peripherals are selected through the peripherals routing controls.

The actual number of channels copied in, and out, is configured in a channel count register. This register defines the stride the DMA uses when moving through its circular buffer. The channel count is limited to a power of 2 numbers, such that 1, 2, 4, or 8 channels can be selected.

The format of the data can be configured as 32-bit, 24-bit, 24-bit packed, 16-bit packed, floating, and raw formats. This format is configured independently in the input and output direction.

The DMA initiates when an Fs pulse is seen from the selected audio source. The DMA copies data from the audio source towards the DSP first and issues a done when this task is completed, followed by copying data from the DSP memory towards the audio peripherals.

Because of this behavior using the DMA introduces an extra sample of latency compared to the time domain path.

Direct Audio IO

All peripheral audio sources have custom TIE instructions to directly import audio data from these sources to the Hi-Fi core. These TIE instructions allow for the lowest latency audio input to the core. Alternatively, all audio sources can also be read over the system bus.

All peripheral audio sinks have customer TIE instructions to directly output audio data to these. The sinks need to select “HiFI TIE output” in their routing controls as their input to enable this. These TIE instructions allow for the lowest latency audio input to the core. Alternatively, all audio sources can also be written over the system bus. The sources need to select “System Bus” in their routing controls as their input to enable this.

BUS Audio IO

All audio peripherals are also memory-mapped to the Hi-Fi core. Audio source data can be read over the bus and audio sinks can be written to. The sinks need to select “System Bus” in their routing controls as their input to enable this.

Debug

A JTAG interface port is included for ease of development and debugging.

FastDSP Memory Access and Safeload

The HiFi 3z DSP can directly read and write to all FastDSP memories when the FastDSP is not running. While the FastDSP core is running the HiFi 3z DSP can read and write to all non-active parameter memory banks. There are five memory locations available that the HiFi 3z DSP can use to update the current bank parameters of a single instruction of the FastDSP. *Table 17* lists the HiFi 3z DSP assembler names for the functions used for safeload.

Table 17. HiFi 3z DSP Safeload to the FastDSP Current Bank

NAME	FUNCTION
FDSP_SL_ADDR	FastDSP safeload instruction number
FDSP_SL_P0	FastDSP Safeload Parameter B0
FDSP_SL_P1	FastDSP Safeload Parameter B1
FDSP_SL_P2	FastDSP Safeload Parameter B2
FDSP_SL_P3	FastDSP Safeload Parameter A1
FDSP_SL_P4	FastDSP Safeload Parameter A2

The functionality of this is the same as the functionality of the FastDSP safeload via the control port (see the *FastDSP Parameter Safeload* section). The parameters are also written to the FastDSP as soon as the frame executes, without needing to write a trigger bit.

Audio Data Handling

The device has two subsystems for processing audio data—a low-latency audio subsystem and a processor subsystem. The device relies on the efficient movement of audio samples within and between these subsystems to enable processing and building applications. The Low-Latency Subsystem is centered on the FDSP and an audio fabric that sends individual samples between blocks with minimal latency. The processor subsystem supports applications run on the HiFi 3z processor where blocks of samples are located in system memories and/or where individual samples can be inserted or pulled directly from the audio fabric.

Low-Latency Audio Subsystem

Within the audio subsystem, the functional blocks that act as sources broadcast their audio samples. The functional blocks that are syncs select from the available sources and match the source's frame rate. This system facilitates the construction of chains of blocks moving data between inputs, through the FDSP for filtering and mixing, and to output interfaces, potentially utilizing sample rate converters between major blocks. From the audio subsystem perspective, the connections to the processor subsystem function the same as any other source or sync by providing or consuming individual samples at their selected frame rate.

Processor Subsystem

The applications running on the HiFi 3z access audio data from either blocks of samples storied in memories, memory-mapped sample FIFOs, or specialty processor instructions that read and write sample registers directly attached to the audio subsystem routing matrix. Processing of the data is triggered by interrupts from the audio subsystem based on when data is available or needed. The choice of access method is driven by application needs for sample granularity and latency requirements.

Data Movement Between Subsystems

- With the Audio DMA's ability to source and sync up to 8 channels on 3 separate engines and access system memories, blocks of data can be stored in memories with limited supervision of the processor. These blocks can comprise interleaved or contiguous blocks connected to a single engine. The engines are independent, so they can vary the sample rate, data organization, and block sizes. This is an ideal choice for applications that process data in blocks. It reduces processor overhead, particularly in the number of interrupts required to send a receive data.
- The Direct IO (TIE) gives the most granular access to audio samples by utilizing processor instructions to give direct access to sample registers within the Audio subsystem. At any point, the processor can set a source register or read the most recent sample generated by any other source. Sample rate synchronization is handled by configuring interrupts to trigger based on an audio component's sample rate or a multiple of the sample rate. The Direct IO is targeted to applications that require the lowest latency and those that require working on individual samples. Using the direct IO for high sample rates or multiple sample rates incurs a larger interrupt latency penalty and therefore must be balanced with the ease of use and sample access latency.
- The Audio Bus Map is indented as a middle ground between the other two access methods. A set of 16 input and output FIFOs act as sources and syncs in the audio subsystem. The Audio Bus Map will collect or transmit multiple samples and multiple samples can be pushed or popped within a single sample period. By having the FIFOs mapped to system memory, the method of writing and reading is simple. The application can either use interrupts generated by the audio bus map FIFO status or use the sample interrupts. The Audio Bus Map should be considered by applications that are interested in small numbers of samples from a given source with some latency tolerance or potentially an application interested in multi-rate processing where the latency of the lower rate is strict and the higher rate can be stored in the FIFO.

Example—Simple ANC with Bluetooth Playback

In this example system, the inputs to the audio subsystem are feedforward and feedback microphones generating 192khz samples and 44.1kHz serial audio data from a Bluetooth SoC (BT). The serial audio data is then sent to the ASRC which up-samples and synchronously outputs a 192kHz stream. These three sources are then used by the FDSP to generate anti-noise and mix in the BT. The output of the FDSP is then routed to the DAC.

At this point, an additional path can utilize the existing sample streams for applications running on the processor subsystem. In this case, there could be two independent threads running on the processor that need access to data from the audio subsystem.

The first thread would conduct scene detection by taking individual samples from the FF mic and comparing the BT audio to what is seen at the FB mic. Utilizing a HIFI_INT_CTRL0 to select the ADC and HIFI_INT_CTRL3 to select every nth sample to generate processor interrupts, the corresponding thread could read the current ADC value, converted to floating point format, via the DirectIO instruction in the HiFi core.

The second thread would select the output of the ASRC and FDSP, using them as inputs to an FDEC pair down-sampling both to 48kHz. The output of the FDEC is then used as the source for the ADMA that stores the set of samples in System memory in a 1.31 format. The ADMA then generates interrupts based on the programmed block size.

Distributed DMA Engines (DDE)

The device uses several distributed DMA engines to automate copying between memories and between the UART, QSPI, and I^{3}C interfaces and memory.

The processor uses Direct Memory Access (DMA) to transfer data within memory spaces or between a memory space and a peripheral. The processor can specify data transfer operations and return to normal processing while the fully integrated DMA controller carries out the data transfers independent of processor activity. The DMA controllers are dispersed throughout the infrastructure, as Distributed DMA Engines (DDE) and connected to the AXI Fabric.

The DDEs can perform transfers between a memory and a peripheral or between one memory and another memory. Two DDEs are used for Memory to Memory DMA (MemDMA). One channel is the source channel, and the second is the destination channel.

The CONFIG register is used to set up DMA parameters and operation modes. Writing CONFIG while DMA is already running causes a DMA error, except for when the EN bit is being written to 0.

Additional documentation on the DDEs can be found in the respective DDE register description section.

Control and Data Ports

The device provides multiple control and data interface ports including an I²C interface, an I³C interface, an SPI interface, a quad SPI (QSPI) interface, a UART interface, and a JTAG interface. Each interface supports either a master mode, slave mode, or both modes of operation. Some interface and operating mode combinations are compatible with either automatic self-boot or external host control, while other combinations can only be operated by local software running on the HiFi 3z Core. *Table 18* provides an overview of each supported interface port.

Table 18. Supported Control and Data Interface Ports

INTERFACE	PINS	MODES	ACCESS
I²C Interface Port	SDA0, SCL0	Slave Mode	Standalone Host Control
	SDA0, SCL0, SDA1, SCL1	Master Mode	Local Software Control
I³C Interface Port	SDA1, SCL1	Master/Slave Mode	Local Software Control
SPI Interface Port	SCLK, MOSI, MISO, $\overline{SS}$	Slave Mode	Standalone Host Control
		Master Mode	Local Software Control
QSPI Interface Port	QSPIM_* (CLK, CS0/1, SDIO_1/2/3/4)	Master Mode	Self-Boot Mode or Local Software Control
UART Interface Port	UART_* (RX, TX, RTS, CTS)	Tx/Rx Modes	Standalone or Local Software
JTAG Interface Port	JTAG_* (TDI, TDO, TCK, TMS, TRST)	Master Mode	Debug Only (See HiFi 3z *Overview*)

I²C/SPI Control Interface Port

The device provides a 4-wire SPI control interface and a 2-wire I²C control interface that share control port pins and by default operate as slave mode interfaces. Both slave interfaces require an external clock source from the host. The device also supports master mode operation for both control interfaces that require a local software driver running on the HiFi 3z core. A secondary I²C master interface can also be implemented using the I³C interface.

The device's default operation is to use the I²C control interface in slave mode. To activate the SPI control interface in slave mode, the $\overline{SS}$ pin must be pulled low three times (toggling high in between each assertion). When in I²C interface mode, the unused control pins (ADDR0 and ADDR1) determine the I²C device address. *Table 19* shows the shared control port pin functions based on the selected control interface operating mode.

Table 19. Multifunction Pin Operation by Control Interface Mode

CONTROL PORT PIN	I²C SLAVE MODE	SPI SLAVE MODE
SCL/SCLK	SCL (I²C Clock Input)	SCLK (SPI Clock Input)
SDA/MISO	SDA (I²C Bidirectional Data)	MISO (SPI Data Output)
ADDR1/MOSI	ADDR1 (I²C Address Input Bit 1)	MOSI (SPI Data Input)
ADDR0/$\overline{SS}$	ADDR0 (I²C Address Input Bit 0)	$\overline{SS}$ (SPI Inverted Slave Select)

Each slave interface can read and write from the device memories and main control registers. All addressable registers can be accessed either individually in each transaction (in single address mode) or sequentially in a single transaction (in burst mode). The first byte (Byte 0) of a control port write contains the 7-bit device address plus the R/$\overline{W}$ bit. The next four bytes (Byte 1 through Byte 4) are the 32-bit sub-address of the memory or register location. All subsequent bytes (starting with Byte 5) contain data (such as register, program, or parameter data). Control registers and bits marked as reserved in the register map/programming guide always read back as 0.

I²C/SPI Control Interface Burst Mode Communication

Burst mode addressing, in which the sub-addresses are automatically incremented at word boundaries, can be used for writing large amounts of data to contiguous memory locations. This increment happens automatically after a single word write unless the transaction is stopped (with a stop condition for I²C, or with $\overline{SS}$ brought high for SPI). The control registers and memory locations vary in width from one byte to five bytes, so the auto-increment feature knows the mapping between sub-addresses and the word length of the destination register or memory location.

If large blocks of data must be downloaded to the DSP cores, the output of the cores can be disabled, new data can be loaded, and the core can then be restarted. This is typically done during the booting sequence at start-up or when loading a new program into memory.

Table 20. I²C/SPI Control Data Word Sizes and Address Ranges

BASE ADDRESS	END ADDRESS	DESCRIPTION
0x0000-0000	0x0001-FFFF	Reserved
0x0002-0000	0x0003-FFFF	HiFi 3z Data RAM 0
0x0004-0000	0x0005-FFFF	HiFi 3z Data RAM 1
0x0006-0000	0x0007-FFFF	HiFi 3z Instruction RAM
0x0008-0000	0x0FFF-FFFF	Reserved
0x1000-0000	0x1009-FFFF	Level 2 Memory
0x100A-0000	0x1FFF-FFFF	Reserved
0x2000-0000	0x2009-FFFF	Level 2 Memory Non-Cached
0x200A-0000	0x3FFF-FFFF	Reserved
0x4000-0000	0x4FFF-FFFF	External QSPI Flash Memory
0x5000-0000	0xEFFF-FFFF	Reserved
0xF000-0000	0xF000-11FC	Main Control Registers
0xF000-11FD	0xF001-FFFF	Reserved
0xF002-0000	0xF002-077F	FastDSP Parameter Memory
0xF002-0780	0xF002-FFFF	Reserved
0xF003-0000	0xF003-01FF	FastDSP State Memory
0xF003-0200	0xF003-FFFF	Reserved
0xF004-0000	0xF004-007F	FastDSP Program Memory
0xF004-0080	0xF004-FFFF	Reserved
0xF005-0000	0xF005-01FF	Audio Bus Mapping
0xF005-0200	0xF00F-FFFF	Reserved
0xF010-0000	0xF01F-FFFF	System Fabric Global Program View
0xF020-0000	0xFFFF-FFFF	Reserved

I²C/SPI Control Interface Memories Access

All memory and control register locations are 32-bits or 4 bytes in width. Each data word occupies a single 32-bit address, and when writing to these memories an entire word (all 4 bytes) must be written for the write to complete and take effect (starting with the lowest address and continuing sequentially). Similarly, a read must begin at the lowest memory address, however, all 4 bytes of a data word need not be read before ending the transaction.

The mapping of bytes over the control interface is where the most significant byte of a memory location is written or read first, and the least significant byte is written or read last. The memories can be read or written in burst mode or single-byte mode if the write and read requirements are met.

I²C Slave Control Interface Operation

The device features an I²C-compatible, 2-wire serial interface. The interface comprises a bidirectional serial data line (SDA) and an input serial clock line (SCL) that facilitate communication between the slave device and the upstream I²C host (also referred to as the I²C master). The I²C interface supports FM+ clock rates up to 1MHz, but for most bus capacitances the SDA_MISO_DRIVE bit must be set to 1 to support these operating speeds.

The ADDR0 and ADDR1 pins set the LSBs of the device's I²C interface address. Therefore, each device can be set to one of four unique addresses, allowing multiple ICs to exist on the same I²C bus without address contention. The 7-bit I²C addresses are shown in *Table 21*.

Table 21. Device I²C Interface Address Selection

ADDR1 (MOSI)	ADDR0 ($\overline{SS}$)	SLAVE ADDRESS
0	0	0x28
0	1	0x29
1	0	0x2A
1	1	0x2B

Each slave device is recognized by a unique 7-bit device address. The I²C address format is shown in *Table 22*. The LSB of the first byte sent from the I²C master and sets either a read or write operation, where logic level 1 corresponds to a read operation, and logic level 0 corresponds to a write operation. An I²C data transfer (transaction) is always terminated by a stop condition.

Table 22. Device I²C Interface Address Format

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	1	0	1	0	ADDR1	ADDR0	R/$\overline{W}$ Bit

Both the SDA and SCL pins must have external pullup resistors (2kΩ typical) on the lines/traces connected to the pins. The pull-up voltage on these signal lines cannot be higher than V_{IOVDD}.

I²C Slave Interface Addressing

Initially, each device on the I²C bus is in an idle state and monitoring the SDA and SCL lines for a start condition and the proper address. The I²C master initiates a data transfer by establishing a start condition, defined by a high-to-low transition on SDA while SCL remains high. A start condition indicates that an address/data stream follows. All devices on the bus respond to the start condition and shift the next eight bits (the 7-bit address plus the R/$\overline{W}$ bit) MSB first.

The device that recognizes the transmitted address responds by pulling the data line low during the ninth clock pulse. This ninth bit is known as an acknowledge bit. All other devices withdraw from the bus at this point and return to the idle condition. The R/$\overline{W}$ bit determines the direction of the data. A Logic 0 on the LSB of the first byte indicates that the master writes information to the peripheral, whereas a Logic 1 indicates that the master reads information from the peripheral (after writing the subaddress and repeating the start address). A data transfer takes place until a stop condition is encountered. A stop condition occurs when SDA transitions from low to high while SCL is held high.

Stop and start conditions can be detected at any stage during the data transfer. If these conditions are asserted out of sequence with normal read and write operations, the device immediately jumps to the idle condition. During a given SCL high period, the user can only issue one-start condition, one-stop condition, or a single-stop condition followed by a single-start condition. A no-acknowledge condition is where the SDA line is not pulled low on the ninth clock pulse on SCL.

If an invalid sub-address is issued by the user, the device issues an acknowledge, but no data write occurs and a read is responded to with zero data. If the highest sub-address location is reached while in write mode, the data for the invalid byte is not loaded to any sub-address register.

I²C Slave Interface Read and Write Operations

Figure 57 shows the format of a single word (4-byte) write operation. The device address is sent first (with the write bit set), and then the 32-bit sub-address is sent next (1 byte at a time). Finally, the data word is written (1 byte at a time). All sub-address locations point to the start of a data word and all 4 bytes must be written sequentially for the write to take effect. Ending the transaction before all 4 bytes are written results in no new values being stored.

Figure 58 shows the format of a burst mode write sequence. In this case, the host may require multiple word writes to multiple sub-addresses. The device automatically increments its sub-address after each data word is written (4 bytes) until a stop condition is sent. The device decodes the sub-address and sets the appropriate auto-increment.

The format of a single word (4 bytes) read operation is shown in *Figure 59*. Note that the first R/$\overline{W}$ bit is 0, indicating a write operation because the sub-address still must be written to set up the internal address pointer. After the device acknowledges the receipt of the sub-address, the master must issue a repeated start command, followed by the chip address byte with the R/$\overline{W}$ set to 1 (read). This causes the SDA to set as an output, and the device begins sending the requested read data back to the master. The master now must respond to every ninth pulse with an acknowledge.

Figure 60 shows the format of a burst mode read sequence. In this case, the device also auto-increments its sub-address after each read until the host sends a no-acknowledge condition (ending the burst read) followed by a stop condition. The device always decodes the sub-address and sets the auto-increment appropriately.

The data bytes written over the I²C interface use little endean format (lowest significant byte first followed by most significant bytes). Data word width is 32 bits wide, so while writing the data into the register little endean format must be used. The 32-bit data is divided into 4 bytes. While writing the data into the register, the lowest significant byte must be written first followed by the remaining 3 bytes with the most significant byte as the last or fourth byte.

Figure 57, *Figure 58*, *Figure 59*, and *Figure 60* use the following abbreviations:

- S is the start bit.
- Sr is the repeated start bit.
- P is the stop bit.
- A is acknowledge (ACK)
- $\overline{A}$ is not acknowledge (NACK)

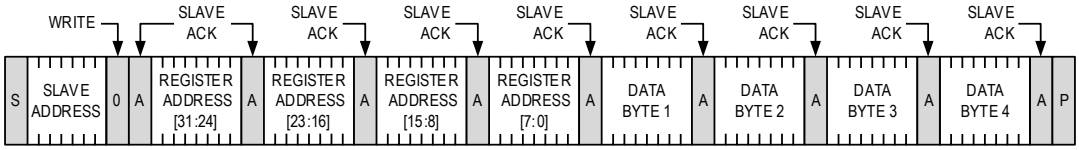

Figure 57. Single-Word I²C Write Format

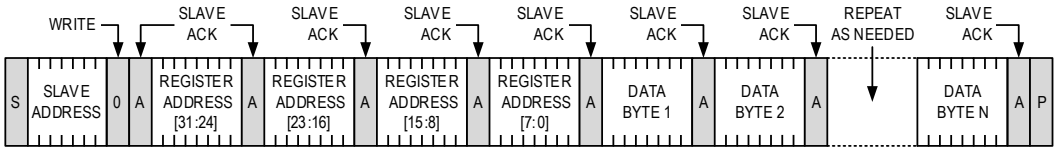

Figure 58. Burst Mode I²C Write Format

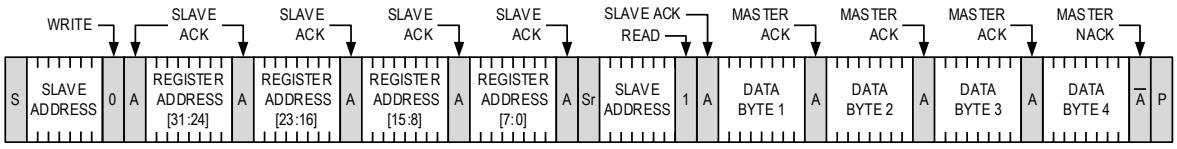

Figure 59. Single-Word I²C Read Format

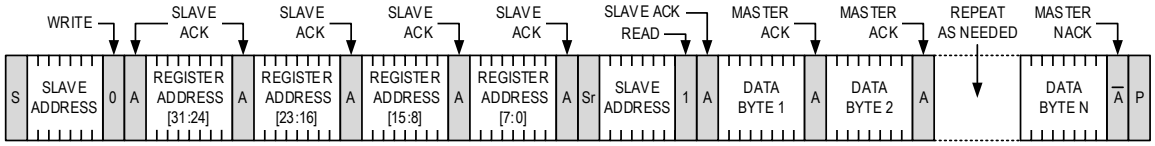

Figure 60. Burst Mode I²C Read Format

SPI Slave Control Interface Operation

By default, the device operates in I²C slave control interface mode. To place the device in SPI slave control interface mode, the $\overline{SS}$ pin must be pulled low three times by issuing three SPI writes (which are in turn ignored by the device). The SPI slave interface is then active, and the device is ready to respond to the next SPI transaction (fourth).

The SPI slave control port is a 4-wire interface consisting of the $\overline{SS}$ SCLK, MOSI, and MISO signals. The $\overline{SS}$ signal must go low at the beginning of a transaction and high at the end of a transaction. The MOSI signal carries the serial input data, and the MISO signal is the serial output data, and all data is sent MSB first. The SCLK signal latches MOSI input data on a rising clock edge. Likewise, MISO output data is changed on the falling SCLK edge and must be clocked into the receiving device (such as a microcontroller) on the rising SCLK edge. The MISO output remains tristated until a read operation is requested, allowing other SPI-compatible peripherals to share the same MISO readback bus.

Once the device is placed into SPI slave interface control mode, it can only be placed back into I²C slave interface control mode by pulling the $\overline{PD}$ pin low (hardware power down) or by powering down the supplies.

SPI Slave Interface Transaction Format

All SPI transactions have the same basic structure, and this format is described in *Table 23*. All device sub-addresses are 32-bits in length (4 bytes) and each data word is also 32-bits wide (4 bytes). As a result, each SPI transaction is at least 10 bytes in length. A timing diagram for SPI transactions is also illustrated in *Figure 5*.

The first byte of an SPI slave interface transaction is always sent on the MOSI line (from master to slave device) and indicates (with the R/$\overline{W}$ bit) whether the transaction is a read or a write. The LSB of this first byte is the R/$\overline{W}$ bit, with a logic level 1 setting a read transaction and a logic level 0 setting a write transaction.

The next 5 bytes (sent on the MOSI line) contain the 32-bit device sub-address location (for the target memory location or control register) followed by an unused dummy byte containing zero code data. The dummy byte effectively extends the sub-address to 40 bits, with the actual sub-address being placed in the 32 MSBs.

For a single read or single write transaction, the next 4 bytes contain the data bits (from MSB to LSB). The device also supports burst mode read and writes, in which case the transaction can be extended by n bytes of data (where n should be a multiple of 4 to read or write n / 4 complete data words). In a burst mode transaction, the device auto-increments the sub-address appropriately. For write transactions, data bytes are received on the MOSI line, while for a read transaction data bytes are transmitted on the MISO line (which is tristated before and after data transmission). The transaction (single or burst mode) is ended when the master pulls the $\overline{SS}$ line high.

Table 23. Generic SPI Interface Transaction Format

BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5	BYTE 6	BYTE 7	BYTE 8	BYTE 9
0000000x (x is the R/$\overline{W}$ Bit)	Target Device Sub-Address				Dummy	First Read or Write Data Word (4 Bytes)			
	Bits[31:24]	Bits[23:16]	Bits[15:8]	Bits[7:0]	00000000	Bits[31:24]	Bits[23:16]	Bits[15:8]	Bits[7:0]

The format for a single-write SPI operation is illustrated in Figure 61 and the format for a single-read SPI operation is shown in *Figure 62*. For simplicity, the $\overline{SS}$ and SCLK lines are not shown but each tick/step on the MOSI or MISO lines represents a single SCLK period. For labels, these diagrams use the following abbreviations:

- SSL is the transaction start where the master pulls the $\overline{SS}$ line low
- SSH is the transaction end where the master pulls the $\overline{SS}$ line low
- X is a don't care condition for the MOSI data input line
- HIGH-Z indicates where the MISO output line is tristated

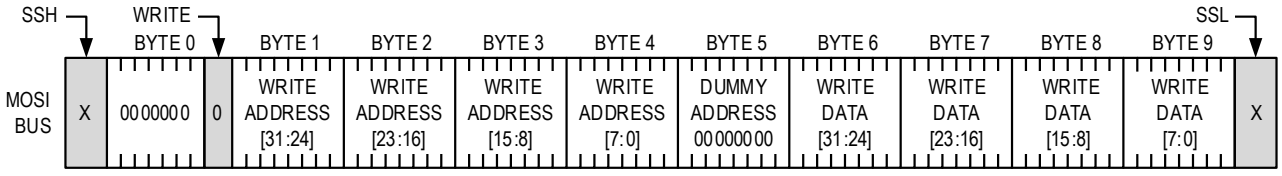

Figure 61. SPI Write Format (Single-Write Mode)

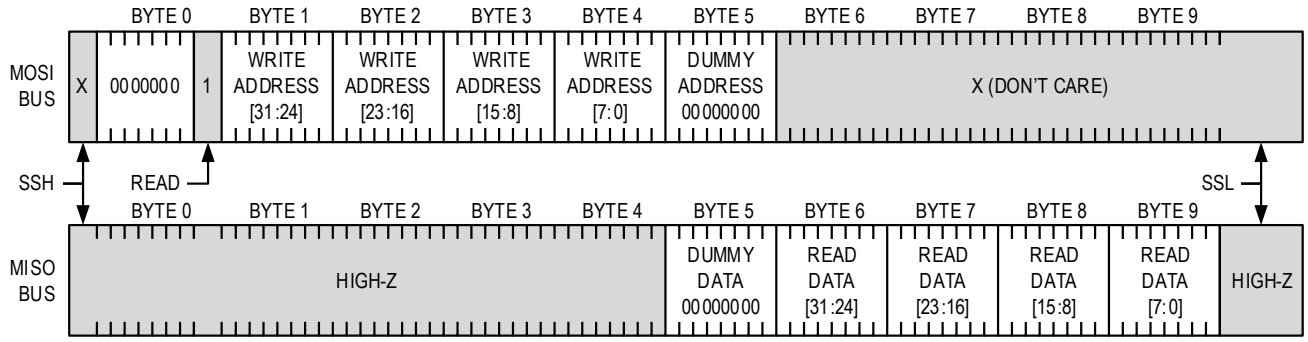

Figure 62. SPI Read Format (Single-Read Mode)

I²C/SPI Master Control Interface Operation

The device can also be configured to support I²C or SPI master control interface modes. This operating mode can only be controlled and configured by a software driver running on the HiFi 3z core. These master mode use cases operate through the same pins as the I²C and SPI slave control interfaces and are mutually exclusive if enabled (the shared pins restrict the device to either an I²C or an SPI interface configured to either master or slave mode).

For the I²C and SPI master control interface modes, the read and write data formats for both single and burst data transfers must be configured to match the address length and word length of the target slave device.

I³C Interface Port

The device also supports an I³C interface port (master or slave mode on SDA1 and SCL1) that can only be controlled and configured by a software driver running on the HiFi 3z core. The I³C interface is compliant with the MIPI I³C v1.1 specification (refer to this documentation for additional details).

The control interface can also be configured to be backward compatible with I²C interfaces, allowing it to support legacy slave devices. All data rates are supported except for the I³C HDR ternary options.

Table 24. Supported I²C/I³C Modes

MODE	SCL FREQ	SDA RATE
I²C Interface Standard Mode	96kHz	96kHz
I²C Fast Mode	384kHz	384kHz
I²C Fast-Mode Plus	768kHz	768kHz
I³C SDR	6.144MHz	6.144MHz
I³C HDR-DDR	3.072MHz	6.144MHz

Quad SPI (QSPI) Master Control Interface Port

The device contains a QSPI master control interface port that is capable of single, dual, or quad data line communication. This port is also used to configure the device from an external memory when in self-boot mode.

The QSPI port can either operate in legacy mode where access and control of the port are through a software driver running on the HiFi 3z core or in memory-mapped mode where the access is automated and external memory can be directly accessed by the host. The host interface contains support for pre-fetch and catching. The QSPI control interface port also has a dedicated DMA channel to automate transfers of data from the port to processor memory.

QSPI Self-Boot Mode Configuration

During initial power-up, the device can either be directly configured by the system host through the I^2C or SPI slave interfaces (referred to as host-boot mode) or it can automatically load the configuration from an EEPROM or Flash through the QSPI master interface (called self-boot mode). The boot mode is selected based on the state of the SELFBOOT input pin, and this is checked internally when the device comes out of reset (exits the hardware full chip power-down state when the supplies are powered up and the $\overline{PD}$ pin is asserted high). If the SELFBOOT input pin is asserted to logic low state then the host-boot mode is selected, while a logic-high state selects self-boot mode.

To place the device into self-boot mode during the initial power-up, the SELFBOOT pin must be asserted to a logic high level either concurrently with or before both the $\overline{PD}$ input pin being asserted high and all supplies being powered up. Likewise, for host-boot mode, the SELFBOOT pin should instead be asserted to a logic-low level. If the use case does not require the SELFBOOT pin to be reconfigured to a multipurpose pin function (MP31 function), the pin can be tied to either IOVDD (for self-boot mode) or DGND (for host-boot mode) through a resistor (typically 10kΩ).

Once the device powers up into self-boot mode an external master clock (MCLK) source must be present to proceed. The default clock settings are configured to accept a master clock frequency of 24.576MHz. If a master clock signal of the correct frequency is not provided, the self-boot operation cannot start (or continue).

Once the self-boot operation begins loading the settings, DVDD must remain stable and within its normal operating range (whether it is internally generated or externally provided). The $\overline{PD}$ input pin must also remain stable and asserted high and the external master clock signal cannot be removed during the self-boot operation.

QSPI Self-Boot Mode Operation

The device's self-boot mode is compatible with an EEPROM or Flash memory that supports a QSPI interface with a 12MHz clock frequency and has a 3-byte address. An example compatible Flash memory (used also on the evaluation board) is the Macronix MX25U3232FM2I02.

During self-boot, if an app pack boot image error is detected and no valid image is found, the self-boot is stopped. The HiFi 3z is then released and ready for operation (see the *Boot Loader* section).

The time to self-boot the device from an EEPROM or Flash memory can be estimated based on the size of the program being loaded (at 12MHz), however, there are some fixed delays to account for as well. The self-boot operation starts after a delay of 16,568 input master clock cycles (from when power-up completes if the master clock is present). With the 24.576MHz master clock, this corresponds to a 0.675ms wait time. If the internal regulator is used (REG_EN is pulled high), then an additional fixed delay of 10ms is added to allow DVDD regulation to come up and settle.

Universal Asynchronous Receiver/Transmitter (UART) Data Interface Port

The UART data interface port is a full-duplex peripheral compatible with PC-style industry-standard UART interfaces. The UART interface converts data between serial and parallel formats. The serial communication follows an asynchronous protocol that supports various word lengths, stop bits, and parity generation options. The UART interface includes interrupt-handling hardware, and interrupts can be generated from multiple events.

In addition to a basic UART operation mode, UART4 supports the half-duplex IrDA® (Infrared Data Association) SIR (9.6/115.2 Kbps rate) protocol and full-duplex Multi-Drop Bus (MDB/ICP v2.0) protocol. The UART interface operating mode is selectable.

Partial modem status and control functionality are supported by the UART module to allow for hardware flow control. The UARTs are DMA-capable peripherals.

The UART interface supports:

- 5 to 8 data bits
- Programmable extra stop bit and programmable extra half stop bit.
- Even, odd, and sticky parity bit options
- 8-stage receive FIFO with programmable threshold interrupt
- Flexible transmit and receive interrupt timings
- Three interrupt outputs for reception, transmission, and status
- Programmable automatic RTS/CTS hardware flow control
- False start bit detection

The UART interface has dedicated DMA channels with support for separate transmitter and receiver DMA master channels. They can be used in either DMA or programmed non-DMA modes of operation. The non-DMA mode requires software management of the data flow using either interrupts or polling. The DMA method requires minimal software intervention as the DMA engine itself moves the data. The UART interface has separate transmit and receive DMA channels, though they may be mixed with other peripherals at the system level. The external peripheral timers can be used to provide a hardware-assisted auto-baud detection mechanism for use with the UART.

Serial Audio Data Interface Ports

The device provides two independently configurable serial audio data interface ports. For each of the two ports (denoted as x), input channels (SDATAI_x) and output channels (SDATAO_x) are enabled with the SPTx_IN_EN and SPTx_OUT_EN bits respectively. A given port is disabled when both input and output channels are disabled. Data is transmitted and received in twos complement (MSB first) format.

The ports each have independent bit clock (BCLK_x) and frame sync clock (FSYNC_x) pins, and each supports operation in both master mode (interface generates bit and frame clock) and slave mode (interface requires an external bit clock and frame clock). The serial audio data interface ports also support common data formats such as I^2S, left-justified, right-justified, and TDM (with up to 16 channels of data). For all support data formats, individual slots (or channels) can be configured to be 16-bit, 24-bit, or 32-bit wide.

Serial Audio Data Interface Format Configuration and Data Routing

The data format for a given serial audio data interface is selected with the SPTx_SAI_MODE bit. When this is set low the serial audio data interface operates in a two-channel (two-slot or stereo) mode. In two-channel modes, both edges of the frame clock determine where data is placed. The left audio data channel is typically mapped to channel or slot 0, and the right audio data channel is typically mapped to channel or slot 1.

The data format in two-channel modes is selected with the SPTx_DATA_FORMAT bits, and can be set to left justified mode (no delay from frame sync clock edges), I^2S mode (1-bit clock of delay from frame sync clock edges), or right justified mode (8-/12-/16-bit clocks of delay from frame sync clock edges). The length of each channel is determined by the external input (slave mode) or configured output (master mode) bit clock to frame sync clock ratio.

When the SPTx_SAI_MODE bit is set high the serial audio data interface operates in TDM mode. In TDM mode the device supports up to 16 data channels (or data slots). In TDM mode, each data channel or data slot can be 16-bits, 24-bits, or 32-bits wide (number of bit clock periods), and this is configured with the SPTx_SLOT_WIDTH bits. The frame sync clock active edge determines where the frame starts (with either no delay or 1-bit clock of delay based on the SPTx_DATA_FORMAT setting). The data in each channel (or slot) is accepted (data input) or placed (data output) sequentially from slot 0 up to a maximum of slot 15 based on the channel (or slot) width.

Table 25. Serial Audio Data Interface Port Supported Data Format Settings

DATA FORMAT	FRAME SYNC CLOCK (SPTx_SAI_MODE)	CHANNEL/SLOT WIDTH (SPTx_SLOT_WIDTH)[1]	DATA DELAY FROM FRAME START (SPTx_DATA_FORMAT)
I^2S Mode	0x0 (50% Duty cycle)	(Don't Care)	0x0 (1-bit clock period of delay)
Left-Justified Mode	0x0 (50% Duty cycle)	(Don't Care)	0x1 (No data delay)
Right-Justified Mode	0x0 (50% Duty cycle)	(Don't Care)	0x2 (8-bit clock periods of delay)
	0x0 (50% Duty cycle)	(Don't Care)	0x3 (12-bit clock periods of delay)
	0x0 (50% Duty cycle)	(Don't Care)	0x4 (16-bit clock periods of delay)
TDM Mode	0x1 (Single Bit Pulse)	16-/24-/32-Bits (0x0/0x1/0x2)	0x1/0x0 (No Delay or 1-bit clock period of delay)

For output data, the device always provides 24-bit data. If the channel/slot width is set to 16-bits, the output data will be truncated. If instead it is set to 32-bits, the 24-bit data is padded out with 8 trailing bits (either zero-padded or high-z). For input data (except for right-justified mode) the port accepts bits sequentially up to a limit of 24. Extra trailing padding bits in an input channel do not cause an error, but extra bits beyond this are ignored (truncated off). The serial port can operate with an arbitrary number of bit clock periods in each frame.

The source data for each output channel is selected with the SPTx_OUT_ROUTEn bits (for Serial port x and output channel n). Valid output data sources include the ASRC output channels, the ADC channels, the digital microphone channels, the decimator channels, the FastDSP output channels, the HiFi 3z TIE output channel, the audio output DMA channels, and the bus map outputs. To designate an audio data output channel as unused or disabled (allowing it to be driven by another device sharing the bus), the SPTx_OUT_ROUTEn bits must be set to 0x7F.

To allow multiple ICs to drive a shared serial audio data output bus, unused serial audio data interface channels (or slots) and any trailing padding bits (with SPTx_SLOT_WIDTH set for 32-bit channels) can be configured to be tri-stated (high-Z). This is configured by setting the SPTx_TRI_STATE bit (disabled by default).

Figure 63, *Figure 64*, and *Figure 65* illustrate two channel use cases (I^2S, left-justified, and right-justified modes).

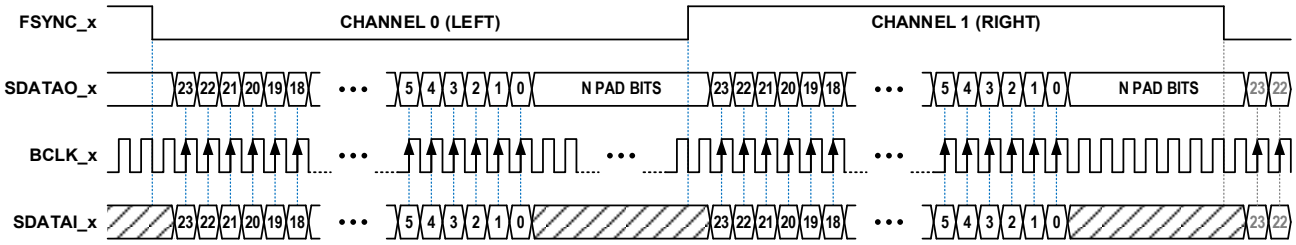

Figure 63. I^2S Mode (1-Bit Clock of Delay, 24-Bit Slot Width, Any Number of Pad Bits Allowed)

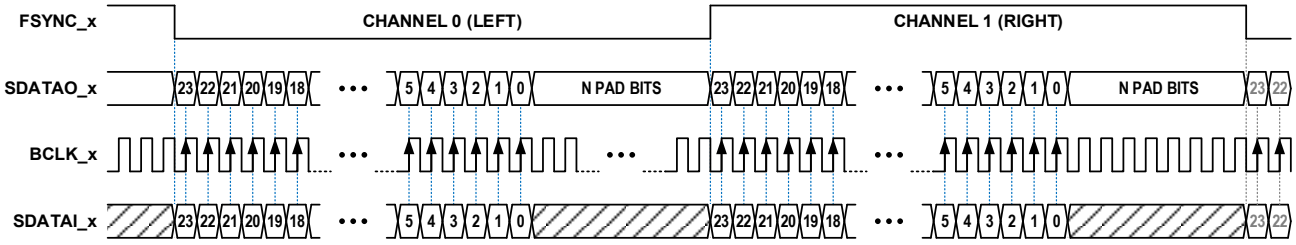

Figure 64. Left-Justified Mode (No Delay, 24-Bit Slot Width, Any Number of Pad Bits Allowed)

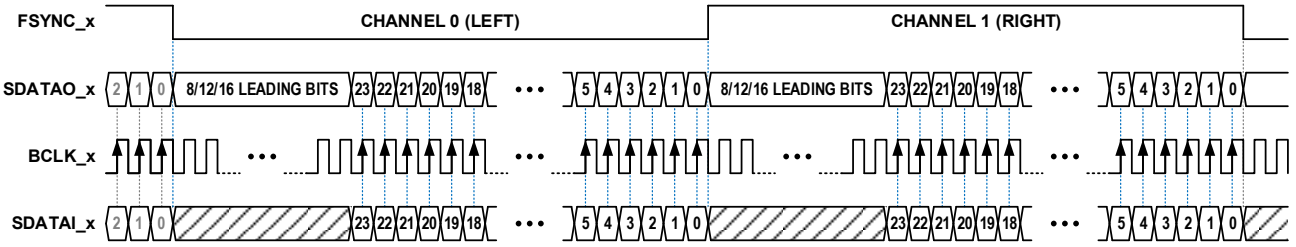

Figure 65. Right-Justified Mode (8-/12-/24-Bit Clocks of Delay, 24-Bit Slot Width)

Figure 66 illustrates a use case where the interface is in TDM mode with timing for 8 channels (or slots).

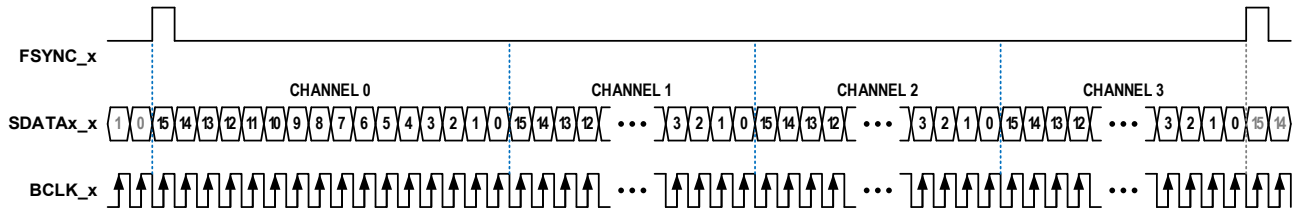

Figure 66. TDM Mode with 4 Audio Data Channels (No Delay, 16-Bit Slot Width, No Padding)

Serial Audio Data Interface Port Clock Configuration

For each of the two serial audio data interfaces, the corresponding FSYNC_x and BCLK_x pins are used to clock both the serial audio data input (SDATAI_x) and output (SDATAO_x) pins. Each port can be configured to operate either in master mode or slave mode.

In master mode, the output bit clock frequency is selected with the SPTx_BCLK_SRC bits, and the frame sync clock frequency is selected with the SPTx_LRCLK_SRC. If the frame sync clock frequency is set to 8kHz or 16kHz, then the selected bit clock frequency is scaled down by 2/3's (Example: a bit clock setting of 3.072MHz becomes 2.048MHz).

To instead place the device in slave mode, both the bit clock frequency setting (SPTx_BCLK_SRC) and the frame sync clock frequency (SPTx_LRCLK_SRC) should be set to the external source setting (0x0 default setting for both). In this mode, an external device or system host must provide both the bit clock and frame sync clock to the configured serial audio data interface. In slave mode, the clocks do not need to be synchronous with the external crystal or master clock input, but the external bit clock and frame sync clock must be synchronous with each other. The input bit clock from either port can also be used as the input clock reference source to the PLL to generate the internal device clocks (instead of a crystal or master clock input).

The active edge polarity of both the bit clock and frame sync clock can be inverted with the SPTx_BCLK_POL and SPTx_LRCLK_POL bits respectively. For example, while serial data and the frame sync clock are by default sampled on the rising edge of the bit clock, setting SPTx_BCLK_POL = 1 inverts this to the falling edge of the bit clock.

When configured for a high bit clock frequency (12.288MHz or higher) in slave mode (clock is an output), it is recommended to increase the drive strength settings for the high-speed output signal pins (in particular the bit clock output and the data output). The high drive strength effectively speeds up the transition times of the waveforms, thereby improving the signal integrity of the clock and data lines. The timing for serial audio data port outputs also changes based on the IOVDD voltage. While the ports can work for inputting a signal on SDATAI_x for any IOVDD and bit clock rate, the drive strength on SDATAO_x at 1.1V excludes operating at higher bit clock rates.

Multipurpose Pin Configuration

The device has 32 multipurpose pins (numbered from MP0 to MP31), each of which is by default configured to its normal function. By using the corresponding MPx_MODE bits, each multipurpose pin can also be set to one of a variety of additional functions including a general-purpose input or output, a master or crystal clock output, a PDM data or clock output, or a system interrupt output.

Care must be taken when using SELFBOOT/MP31 as a multipurpose pin. The state of this pin at power-up (later of either $\overline{PD}$ pin going high or power being applied with $\overline{PD}$ pin already high) determines whether the device self-boots, which must still be followed even if the pin is later reprogrammed to another multipurpose function.

When an MPx pin is set as a general-purpose input, the pin input state can be read by any control interface through the corresponding GPIOx_IN bit. A general-purpose input can also be used by the HiFi 3z core as an external interrupt source by using the level and edge interrupts. In addition, a general-purpose input pin can be used by the FastDSP to conditionally execute instructions or trigger the compressor.

When an MPx pin is set as a general-purpose output, the pin output state can be set by any control interface through the corresponding GPIOx_OUT bit.

Any MPx pin can be configured as either a direct crystal output or as a master clock output. When configured as a master clock output, the frequency of the output is divided down by an integer ratio from the internally generated master clock. The master clock divider ratio is selected with the MCLKO_RATE bits. By default, on initial device power-up, the SELFBOOT/MP31 pin is configured to automatically output the divided down master clock, however, multiple pins can be assigned to this same function if required by the use case.

Any MPx pin can be used to output the PDM clock or data signal for the PDM output interface. In this case, the multipurpose pin can be used to route the PDM data and clock to an external amplifier or upstream host device.

Finally, any MPx pin can be configured to output the interrupt status from any of the nine system interrupt sources (IRQ1 through IRQ9). This is described in more detail in the *Multipurpose Pins as Interrupt Outputs* section.

When using the MPx pins as a high-speed output (such as a master clock, crystal clock, PDM clock, or PDM data line), care must be taken to minimize external capacitive loading on the multipurpose pin otherwise there could be a significant increase in IOVDD current.

Table 26. Multipurpose Pin Functions

MPx PIN FUNCTION[1]	DIRECTION
General-Purpose Input (GPI) to GPIOx_IN Bits, HiFi 3z Interrupts, or FastDSP Triggers	In
General-Purpose Output (GPO) from GPIOx_OUT Bits	Out
Integer Divided Master Clock (MCLK) Output as set by the MCLKO_RATE Bits	Out
PDM Clock Output (PDM_CLK)	Out
PDM Data Output (PDM_DAT)	Out
System IRQx Output (Where x is IRQ1 through IRQ9)	Out

[1] These functions are selected instead of the pins default function with the MPx_MODE bits.

Multipurpose Pins as Interrupt Outputs

Each multipurpose pin can be used to output one of the nine system interrupt bus channels (from IRQ1 to IRQ9), and each of these channels can have a different combination of unmasked individual interrupt sources. The output of each system interrupt bus channel can be individually inverted in polarity with the corresponding IRQx_INVERT bit. Refer to the device main register map programming guide for a full list of the individual interrupt sources.

Each interrupt source has its own individual status bit (IRQ_x) and a clear bit (IRQ_x_CLR). The status of each interrupt source is read with the corresponding IRQ status bit (IRQ_x). Once an interrupt status bit is set (IRQ_x) it latches and will remain set (even if that interrupt source is no longer asserted/true) until cleared by setting with the corresponding interrupt clear bit (IRQ_x_CLR).

Each interrupt source has nine mask bits (IRQn_x_MASK) where n is the system interrupt channel (from IRQ1 to IRQ9) and where x is the individual interrupt source. Each system interrupt channel (IRQn) then represents one combined interrupt signal (comprised of all IRQn unmasked individual interrupt sources) that can be assigned to a multipurpose pin. These system interrupt channels are shared by the cores as interrupt options.

Multipurpose Pin Level Control Options

Each digital pin that has both a default function and multipurpose functions has a corresponding control register (x_CTRL where x describes the default function, for example, BCLK0_CTRL for the BCLK_0/MP2 pin). The bits within these registers are used to set pin-level parameters such as weak pull-up/down, slew rate, and drive strength. The pin control settings affect the operation in both default function mode and when used in multipurpose pin modes.

When a multipurpose pin is used as an output, the drive strength can be set to 2mA, 4mA, 8mA, or 12mA with the corresponding x_DRIVE bits. In addition, when used as an output the slew rate can be set to either fast mode or slow mode with the corresponding x_SLEW bit.

If a weak pull-up or pull-down is required for a multipurpose pin (when used as either an input or output), this function is selected with the corresponding x_PULL_SEL bit. The selected pull-up or pull-down function can then be toggled (enabled or disabled) with the corresponding x_PULL_EN bit.

APPLICATIONS INFORMATION

Power Supply Decoupling Capacitors

Bypass each analog and digital power supply pin to its nearest appropriate ground pin with a single 0.1µF capacitor. The connections to each side of the capacitor must be as short as possible, and the trace must be routed on a single layer with no vias. For maximum effectiveness, locate the capacitor equidistant from the power and ground pins or slightly closer to the power pin if the equidistant placement is not possible. Thermal connections to the ground planes must be made on the far side of the capacitor.

Each supply signal on the board must also be bypassed with a single bulk capacitor (2.2µF).

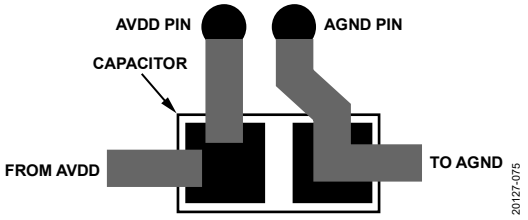

Figure 67. Recommended Power Supply Bypass Capacitor Layout

Layout

The HPVDD supply is for the headphone amplifiers. If the headphone amplifiers are enabled, the PCB trace to this pin must be wider than the traces to other pins to increase the current carrying capacity. A wider trace must also be used for the headphone output lines.

Grounding

Use a single ground plane in the application layout. Place the components in the analog signal path away from the digital signals.

OUTLINE DIMENSIONS

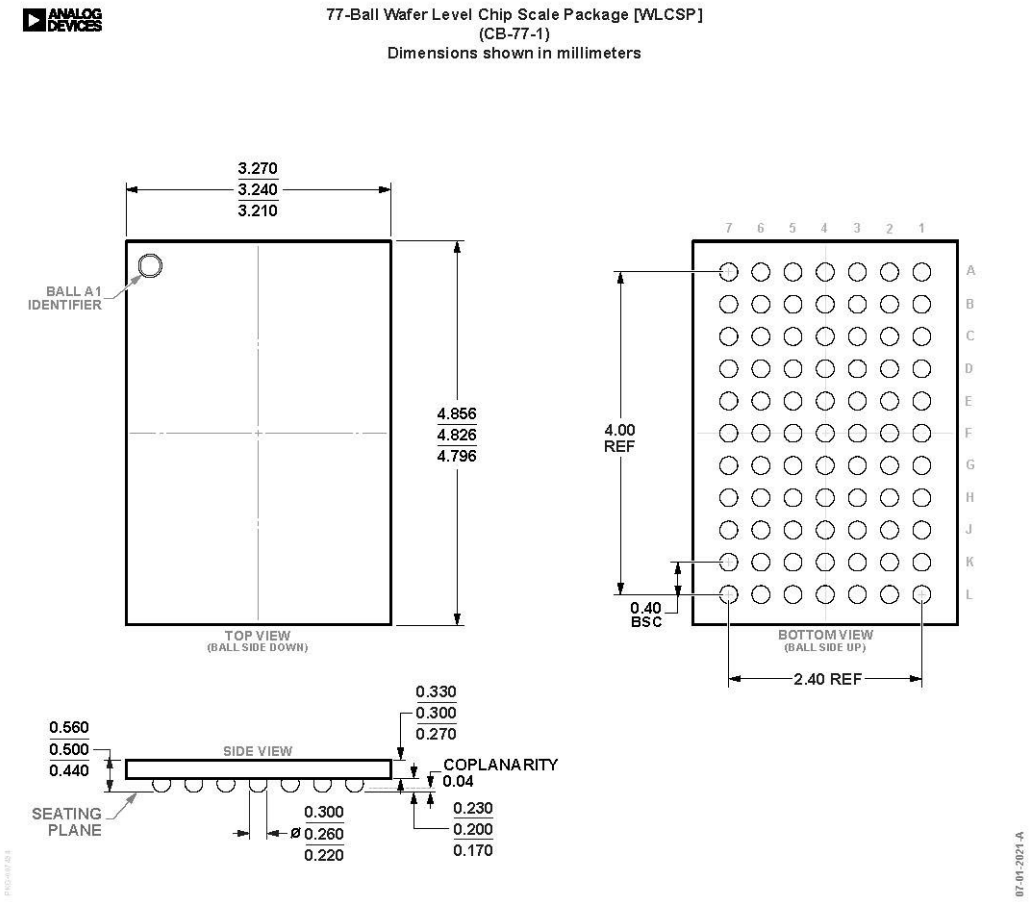

Figure 68. 77-Ball Wafer Level Chip Scale Package [WLCSP]
(CB-77-1)
Dimensions Shown in Millimeters

ORDERING GUIDE

Table 27. Ordering Guide

MODEL[1]	TEMPERATURE RANGE	PACKAGE DESCRIPTION	PACKAGE OPTION
ADAU1797BCBZRL	-40°C to +85°C	77-Ball Wafer Level Chip Scale Package [WLCSP]	CB-77-1
EVAL-ADAU1797Z		Evaluation Board	

[1] Z = RoHS Compliant Part.