SigmaDSP® 28/56-Bit Audio Processor with 2ADC/4DAC

Preliminary Technical Data

ADAU1701

FEATURES

28/56-bit, 50 MHz digital audio processor
Stereo ADC: 100 dB SNR and -80 dB THD+N
4-channel DAC: 104 dB SNR and -90 dB THD+N
Complete stand-alone operation
- **Self-boot from serial EEPROM**
- **Auxiliary ADC with four-input mux for analog control**
- **GPIOs for digital controls and outputs**

Fully programmable with SigmaStudio™ graphical tool
28-bit × 28-bit multiplier with 56-bit accumulator for full double precision processing
Clock Oscillator for generating master clock from crystal
PLL for generating master clock from 64 × f_S, 256 × f_S, 384 × f_S, or 512 × f_S clocks
Flexible serial data I/O ports with I²S compatible, left-justified, right-justified, and TDM modes
Sampling rates up to 192 kHz supported
On-chip voltage regulator for compatibility with 3.3 V systems
48-lead LQFP plastic package

APPLICATIONS

Multimedia speaker systems
MP3 player speaker docks
Automotive head units
Mini-component stereos
Digital televisions
Studio monitors
Speaker crossovers
Musical instrument effects processors
In-seat sound systems (aircraft/motor coaches)

FUNCTIONAL BLOCK DIAGRAM

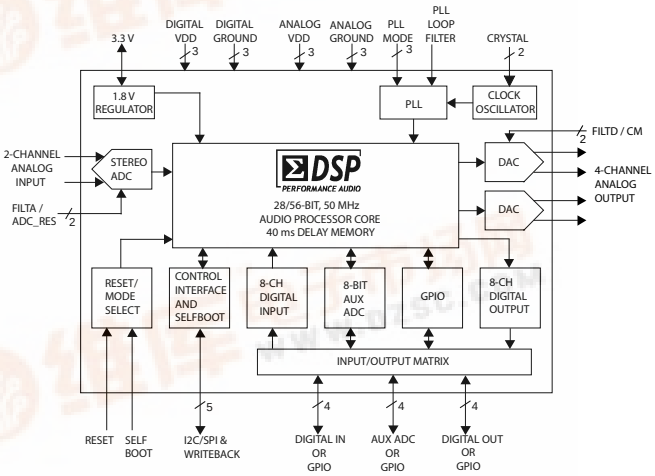


Figure 1.

GENERAL DESCRIPTION

The ADAU1701 is a complete audio system-on-a-chip including a 28/56-bit audio DSP, ADCs and DACs, and microcontroller-like control interfaces. Signal processing includes equalization, crossover, bass enhancement, multiband dynamics processing, delay compensation, speaker compensation, and stereo image widening. These algorithms can be used to compensate for the real-world limitations of speakers, amplifiers, and listening environments, resulting in a dramatic improvement of perceived audio quality.

The signal processing used in the ADAU1701 is comparable to that found in high end studio equipment. Most of the processing is done in full 56-bit double-precision mode, resulting in very good low level signal performance. The ADAU1701 is a fully-programmable DSP. The easy-to-use SigmaStudio software allows the user to graphically configure a custom signal processing flow using blocks such as biquad filters, dynamics processors, level controls, and GPIO interface controls.

ADAU1701 programs can be loaded on power-up either from a serial EEPROM though its own self-boot mechanism or from an external microcontroller. On power-down, the current state of the parameters can be written back to the EEPROM from the ADAU1701 to be recalled the next time the program is run.

The ADAU1701's two sigma-delta (Σ-Δ) ADCs and four Σ-Δ DACs provide an analog-in to analog-out dynamic range greater than 100 dB. The ADC's THD+N is -80 dB and the DAC's is -90 dB. Digital input and output ports allow a glueless connection to additional ADCs and DACs. The ADAU1701 communicates through either an I²C bus or a 4-wire SPI port.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781.329.4700 www.analog.com

TABLE OF CONTENTS

REVISION HISTORY

4/06—Preliminary Version PrF

SPECIFICATIONS

Table 1. Test conditions, unless otherwise noted

Parameter	Conditions
Analog Supply Voltage (AVDD)	3.3 V
Digital Supply Voltage (DVDD)	1.8 V
PLL Voltage (PVDD)	3.3 V
Input/Output Voltage (IOVDD)	3.3 V
Ambient Temperature	25° C
Master Clock Input	12.288 MHz
Load Capacitance	
Load Current	
Input Voltage, HI	
Input Voltage, LO	

ANALOG PERFORMANCE

Table 2. Analog Performance

Parameter	Min	Typical	Max	Units	Test Conditions/Comments
VOLTAGE REFERENCE					
Absolute Voltage (CM, FILTA, FILTD)		1.5		V	
Temperature Coefficient		130		ppm/°C	
AUX ANALOG INPUTS					
Full Scale Analog Input		3.3		V	
INL		TBD			
DNL		TBD			
Offset		TBD		mV	
Input impedance		30		kΩ	
ADC INPUTS					
Number of channels		2			Stereo input
Resolution		24		Bits	
Full Scale Input		100 (283)		μA_{rms} (µA p-p)	$2V_{rms}$ input with 20kΩ (18 kΩ external + 2 kΩ internal) series resistor
Signal-to-Noise Ratio					
A-Weighted		100		dB	
Dynamic Range					-60dB with respect to full scale Analog input
A-Weighted		100		dB	
Total Harmonic Distortion + Noise		-80		dB	-1 dB with respect to full scale Analog input
Interchannel Gain Mismatch		TBD		dB	Left and Right channel Gain Mismatch
Crosstalk		TBD		dB	Analog channel-to-channel crosstalk
DC Bias		1.5		V	
Gain Error		TBD		%	
Power Supply Rejection		TBD		dB	1kHz, $300mV_{P-P}$ Signal at AVDD
DAC OUTPUTS					
Number of channels		4			2 stereo output channels
Resolution		24		Bits	
Full Scale Analog Output		0.9 (2.5)		V_{rms} (V pp)	
Signal-to-Noise Ratio					
A-Weighted		104		dB	
Dynamic Range					-60dB with respect to full scale Analog input
A-Weighted		104		dB	
Total Harmonic Distortion + Noise		-90		dB	-1 dB with respect to full scale Analog input
Crosstalk		TBD		dB	Analog channel-to-channel crosstalk
Interchannel Gain Mismatch		TBD		dB	Left and Right channel Gain Mismatch

Parameter	Min	Typical	Max	Units	Test Conditions/Comments
Gain Error		TBD		%	
DC Bias		1.5		V	
Power Supply Rejection		TBD		dB	1kHz, 300mV$_{P-P}$ Signal at AVDD

DIGITAL I/O

Table 3. Digital I/O

Parameter	Min	Typ	Max	Unit
Input Voltage, HI (V_{IH})	2.0		IOVDD	V
Input Voltage, LO (V_{IL})			0.8	V
Input Leakage, HI (I_{IH})			10	µA
Input Leakage, LO (I_{IL})			10	µA
High Level Output Voltage (V_{OH}), I_{OH} = 1 mA	2.0			V
Low Level Output Voltage (V_{OL}), I_{OL} = 1 mA			0.8	V
Input Capacitance			5	pF
GPIO Output Drive		5		mA

POWER

Table 4.

Parameter	Comments	Min	Typ	Max[1]	Unit
Supply Voltage					
Analog Voltage			3.3		V
Digital Voltage			1.8		V
PLL Voltage			3.3		V
IOVDD Voltage			3.3		V
Supply Current					
Analog Current (AVDD & PVDD)			65	85	mA
Digital Current (DVDD)			40	60	mA
PLL Current			TBD		mA
Analog Current, Reset			TBD		mA
Digital Current, Reset			TBD		mA
PLL Current, Reset			TBD		mA
Dissipation					
Operation (AVDD, DVDD, PVDD)[2]			286.5		mW
Reset, all supplies			TBD		mW

[1] Maximum specifications are measured across −xx°C to xx°C (case) and across VDD = xxx V to xxx V.
[2] Power dissipation does not include IOVDD power because the current draw from this supply is dependant on loads on the digital output pins.

TEMPERATURE RANGE

Table 5.

Parameter	Min	Typ	Max	Unit
Functionality Guaranteed	0°C		70°C	°C Ambient

DIGITAL TIMING

Table 6 Digital Timing[1]

Parameter		Comments	Min	Max	Unit
MASTER CLOCK					
t_{MP}	MCLK Period	512 f_S mode	36	244	ns
t_{MP}	MCLK Period	384 f_S mode	48	366	ns
t_{MP}	MCLK Period	256 f_S mode	73	488	ns
t_{MP}	MCLK Period	64 f_S mode	291	1953	ns
SERIAL PORT					
t_{BIL}	INPUT_BCLK LO Pulse Width		40		ns
t_{BIH}	INPUT_BCLK HI Pulse Width		40		ns
t_{LIS}	INPUT_LRCLK Setup	To INPUT_BCLK rising	10		ns
t_{LIH}	INPUT_LRCLK Hold	From INPUT_BCLK rising	10		ns

t_{SIS}	SDATA_INx Setup	To BCLK_IN rising	10		ns
t_{SIH}	SDATA_INx Hold	From BCLK_IN rising	10		ns
t_{LOS}	OUTPUT_LRCLK Setup	Slave mode	10		ns
t_{LOH}	OUTPUT_LRCLK Hold	Slave mode	10		ns
t_{TS}	OUTPUT_BCLK Falling to OUTPUT_LRCLK Timing Skew				ns
t_{SODS}	SDATA_OUTx Delay	Slave mode, from OUTPUT_BCLK falling		40	ns
t_{SODM}	SDATA_OUTx Delay	Master mode, from OUTPUT_BCLK falling		40	ns
SPI PORT					
t_{CCPL}	CCLK Pulse Width LO		TBD		ns
t_{CCPH}	CCLK Pulse Width HI		TBD		ns
t_{CLS}	CLATCH Setup	To CCLK rising	TBD		ns
t_{CLH}	CLATCH Hold	From CCLK rising	TBD		ns
t_{CLPH}	CLATCH Pulse Width HI		TBD		ns
t_{CDS}	CDATA Setup	To CCLK rising	TBD		ns
t_{CDH}	CDATA Hold	From CCLK rising	TBD		ns
t_{COD}	COUT Delay	From CCLK rising		TBD	ns
I^2C PORT					
f_{SCL}	SCL Clock Frequency			400	kHz
t_{SCLH}	SCL High		0.6		µs
t_{SCLL}	SCL Low		1.3		µs
t_{SCS}	Setup Time	Relevant for Repeated Start Condition	0.6		µs
t_{SCH}	Hold Time	After this period the 1st clock is generated	0.6		µs
t_{DS}	Data Setup Time		100		ns
t_{SCR}	SCL Rise Time			300	ns
t_{SCF}	SCL Fall Time			300	ns
t_{SDR}	SDA Rise Time			300	ns
t_{SDF}	SDA Fall Time			300	ns
t_{BFT}	Bus-Free Time	Between Stop and Start	0.6		
MULTIPURPOSE PINS & RESET					
t_{GRT}	GPIO Rise Time			TBD	ns
t_{GFT}	GPIO Fall Time			TBD	ns
t_{GIL}	GPIO Input Latency	Until high/low value read by core		1.5 × 1/fs	µs
t_{RLPW}	RESETB LO Pulse Width		20		ns

[1] All timing specifications are given for the default (I^2S) states of the serial input control port and the serial output control ports. See Table 45.

PLL

Table 7.

Parameter	Min	Typ	Max	Unit
Operating Range	TBD		TBD	MHz
Lock Time			20	ms

REGULATOR[1]

Table 8.

Parameter	Min	Typ	Max	Unit
DVDD Voltage		1.8		V

[1] Regulator specifications are calculated using an FZT953 transistor in the circuit.

ABSOLUTE MAXIMUM RATINGS

Table 9.

Parameter	Min	Max	Unit
DVDD to GND	0	2.2	V
AVDD to GND	0	4.0	V
IOVDD to GND	0	4.0	V
Digital Inputs	DGND – 0.3	IOVDD + 0.3	V
Maximum Junction Temperature		135	°C
Storage Temperature Range	–65	+150	°C
Soldering (10 sec)		300	°C

Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Table 10. Package Characteristics

Parameter	Min	Typ	Max	Unit
θ_{JA} Thermal Resistance (Junction-to-Ambient)		72		°C/W
θ_{JC} Thermal Resistance (Junction-to-Case)		19.5		°C/W

ESD CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although this product features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

TYPICAL PERFORMANCE CHARACTERISTICS

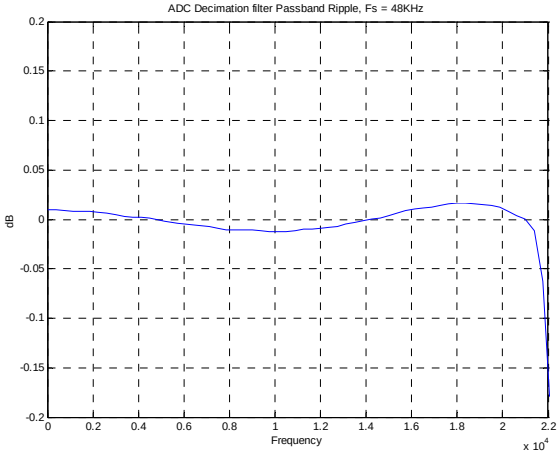


Figure 2. ADC Passband Filter Response, fs=48 kHz

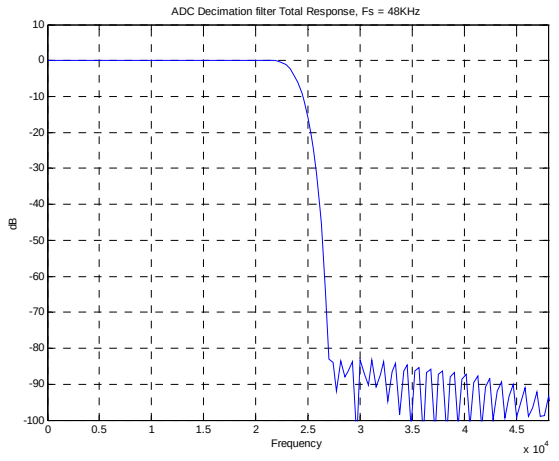


Figure 3. ADC Stopband Filter Response, fs=48 kHz

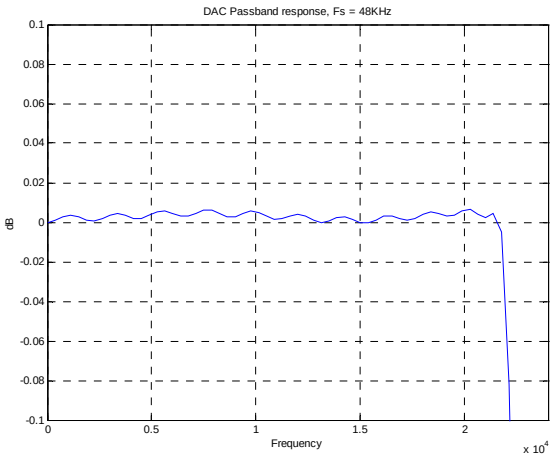


Figure 4. DAC Passband Filter Response, fs=48 kHz

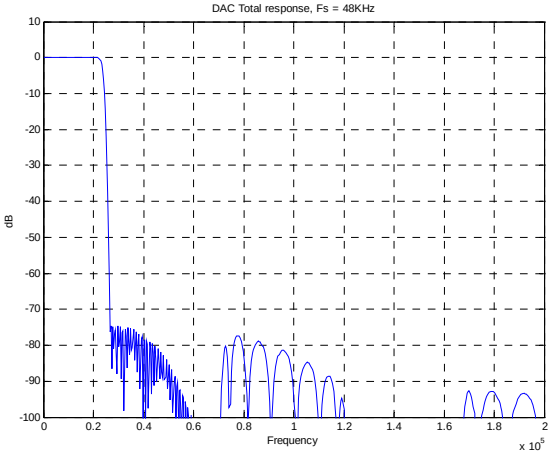


Figure 5. DAC Stopband Filter Response, fs=48 kHz

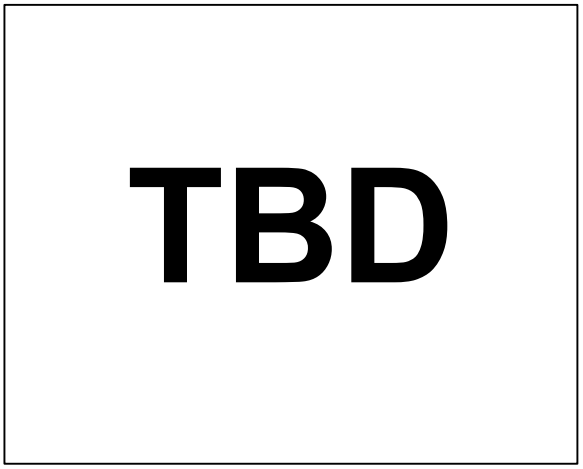


Figure 6. ADC frequency response with 1 kHz, -60 dBFS input

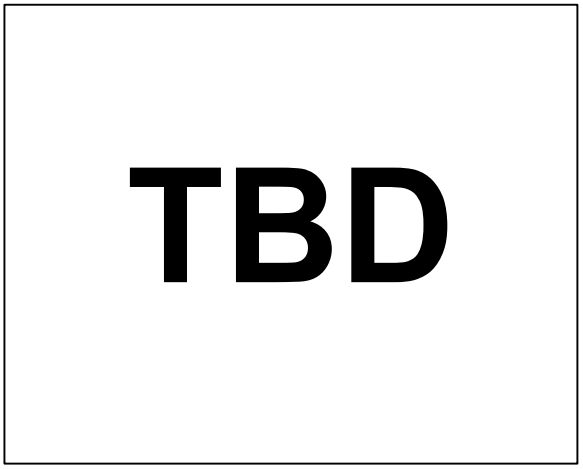


Figure 7. ADC frequency response with 1 kHz, 0 dBFS input

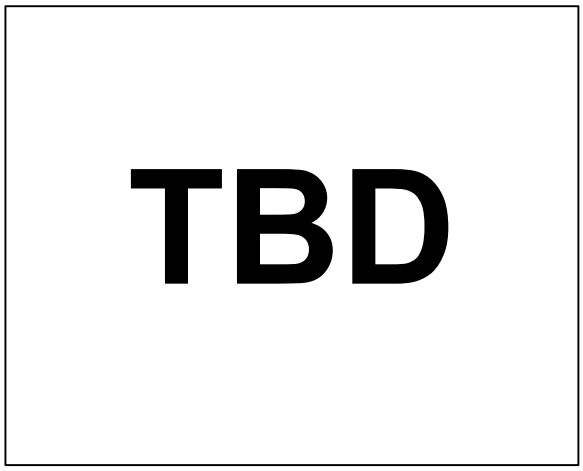


Figure 8. ADC Total Harmonic Distortion + Noise vs. frequency

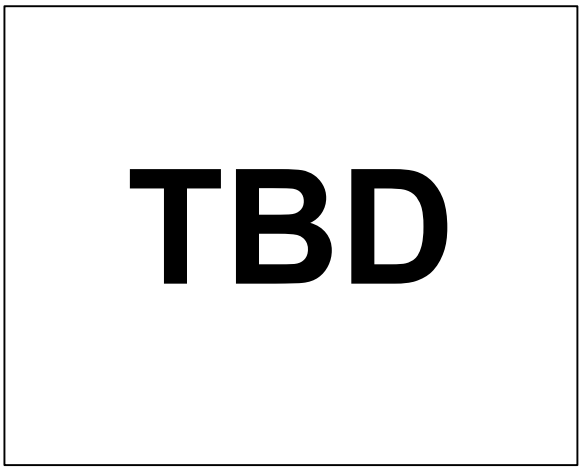


Figure 11. DAC Total Harmonic Distortion + Noise vs. frequency

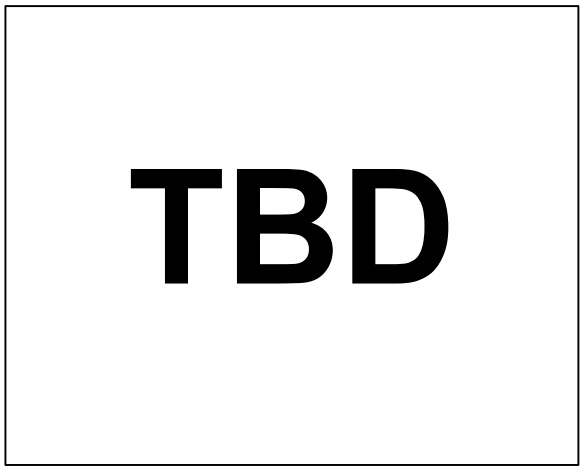


Figure 9. DAC frequency response with 1 kHz, -60 dBFS input

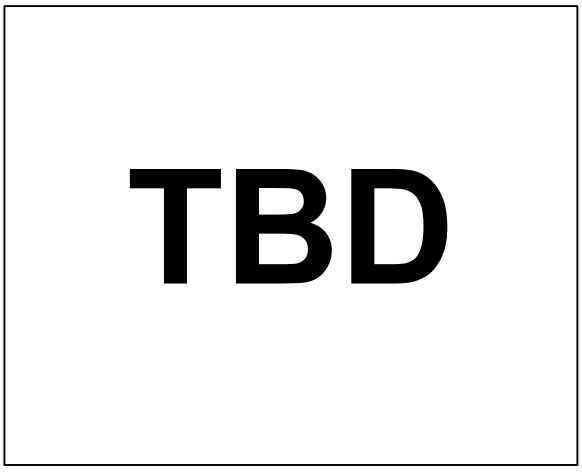


Figure 12. Crosstalk vs. frequency

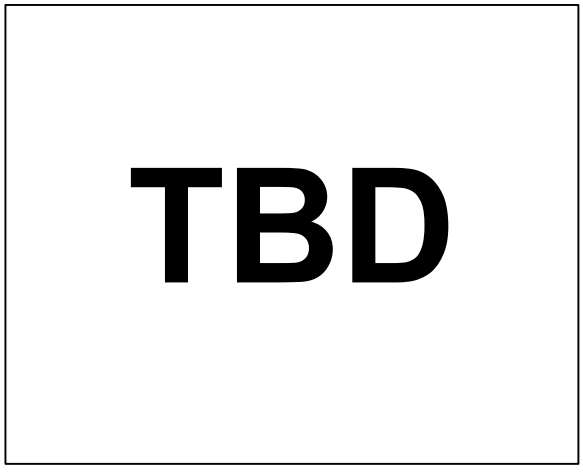


Figure 10. DAC frequency response with 1 kHz, 0 dBFS input

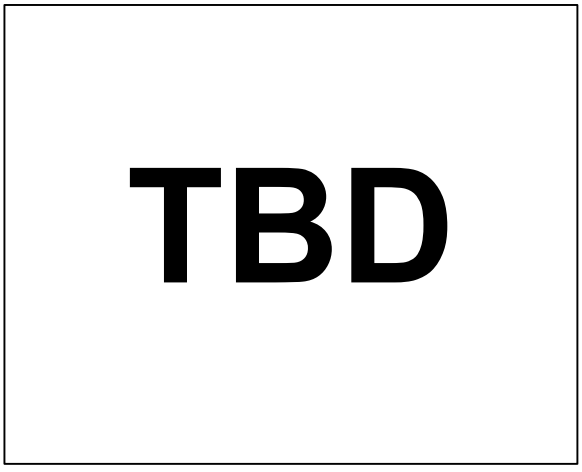


Figure 13. PSRR vs. frequency

PIN CONFIGURATION AND FUNCTIONS

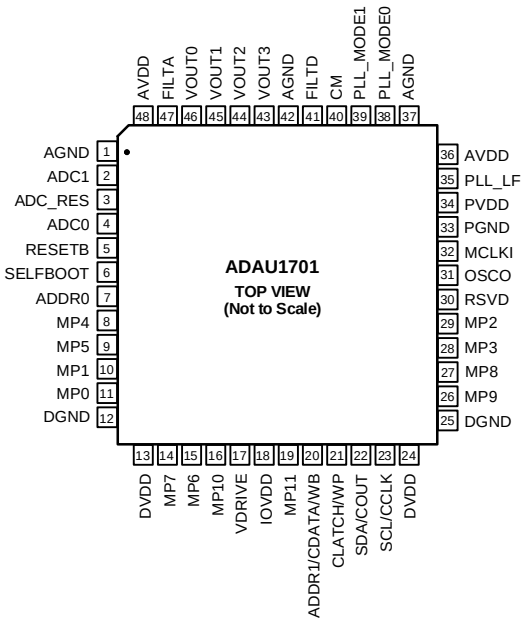


Figure 14. 48-Lead LQFP Pin Configuration

Table 11. Pin Function Descriptions

Pin No.	Type[1]	Mnemonic	Description
1	PWR	AGND	AGND is an analog ground pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. AGND should be decoupled to an AVDD pin with a 100 nF capacitor.
2	A_IN	ADC1	Analog input 1, full-scale 100 µA rms input. Current input allows input voltage level to be scaled with an external resistor.
3	A_IN	ADC_RES	ADC reference current. The full-scale current of the ADCs can be set with an external resistor connected between this pin and ground.
4	A_IN	ADC0	Analog Input 0, full-scale 100 µA rms input. Current input allows input voltage level to be scaled with an external resistor.
5	D_IN	RESETB	RESETB is an active-low reset input. Reset is triggered on a high-to-low edge and the part will exit reset on a low-to-high edge. For detailed information about initialization, see the Power-Up Sequence section.
6	D_IN	SELFBOOT	SELFBOOT selects control port (0) or self-boot (1). Setting this pin high will initiate a selfboot operation when the ADAU1701 is brought out of reset. This pin can be tied directly to the control voltage or pulled up/down with a resistor.
7	D_IN	ADDR0	I^2C and SPI Address 0, in combination with ADDR1will allow up to four ADAU1701s to be used on the same I^2C bus and up to two ICs to be used with a common SPI CLATCH signal.
8	D_IO	MP4	Multi-Purpose – GPIO or Serial input port LRCLK (INPUT_LRCLK)
9	D_IO	MP5	Multi-Purpose – GPIO or Serial input port BCLK (INPUT_BCLK)
10	D_IO	MP1	Multi-Purpose – GPIO or Serial Input port data 1 (SDATA_IN0)
11	D_IO	MP0	Multi-Purpose – GPIO or Serial Input port data 0 (SDATA_IN1)
12	PWR	DGND	DGND is a digital ground pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. DGND should be decoupled to a DVDD pin with a 100 nF capacitor.
13	PWR	DVDD	1.8 V Digital Supply. This can be supplied either externally or generated from a 3.3 V supply with the on-board 1.8 V regulator. DVDD should be decoupled to

			DGND with a 100 nF capacitor.
14	D_IO	MP7	Multi-Purpose – GPIO or Serial output port data 1 (SDATA_OUT1)
15	D_IO	MP6	Multi-Purpose – GPIO, Serial output port data 0, or TDM data output (SDATA_OUT0)
16	D_IO	MP10	Multi-Purpose – GPIO or Serial output port LRCLK (OUTPUT_LRCLK)
17	A_OUT	VDRIVE	Drive for 1.8 V regulator. The base of the voltage regulator's external PNP transistor is driven from VDRIVE.
18	PWR	IOVDD	Input and Output pins supply. The voltage on this pin sets the highest input voltage that should be seen on the digital input pins. This pin is also the supply for the digital output signals on the control port and MP pins. IOVDD should always be set to 3.3 V. The current draw of this pin is variable because it is dependant on the loads of the digital outputs.
19	D_IO	MP11	Multi-Purpose – GPIO or Serial output port BCLK (OUTPUT_BCLK)
20	D_IN	ADDR1/CDATA/WB	ADDR1: I^2C Address 1, in combination with ADDR0 it will set the I^2C address of the IC. Four ADAU1701s to be used on the same I^2C bus. CDATA: SPI Data Input WB: EEPROM Writeback trigger. A rising (default) or falling (if set in the EEPROM messages) edge on this pin will trigger a write-back of the interface registers to the external EEPROM. This function can be used to save parameter data on power-down.
21	D_IN	CLATCH / WP	CLATCH: This SPI latch signal must go low at the beginning of an SPI transaction and high at the end of a transaction. Each SPI transaction may take a different number of CCLKs to complete, depending on the address and read/write bit that are sent at the beginning of the SPI transaction. WP: Self-boot EEPROM write protect. This pin is an open collector output when in selfboot mode. The ADAU1701 will pull this low to prohibit writes to an external EEPROM. This pin should be pulled high to 3.3 V.
22	D_IO	SDA/COUT	SDA: This I^2C data pin is a bidirectional open collector. The line connected to this pin should have a 2.2 kΩ pull-up resistor. COUT: This SPI data output is used for reading back registers and memory locations. It is three-stated when an SPI read is not active.
23	D_IO	SCL/CCLK	SCL: This I^2C clock pin is always an open collector input when in I^2C control mode. In self-boot mode this pin will be an open collector output (I^2C master). The line connected to this pin should have a 2.2 kΩ pull-up resistor. CCLK: This SPI clock may either run continuously or be gated off in between SPI transactions.
24	PWR	DVDD	1.8 V Digital Supply. This can be supplied either externally or generated from a 3.3 V supply with the on-board 1.8 V regulator. DVDD should be decoupled to DGND with a 100 nF capacitor.
25	PWR	DGND	DGND is a digital ground pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. dGND should be decoupled to a DVDD pin with a 100 nF capacitor.
26	D/A_IO	MP9	Multi-Purpose – GPIO, serial output port data 3 (SDATA_OUT3), or auxiliary ADC input 0
27	D/A_IO	MP8	Multi-Purpose – GPIO, serial output port data 2 (SDATA_OUT2), or auxiliary ADC input 3
28	D/A_IO	MP3	Multi-Purpose – GPIO, serial input port data 3 (SDATA_IN3), or auxiliary ADC input 2
29	D/A_IO	MP2	Multi-Purpose – GPIO, serial input port data 2 (SDATA_IN2), or auxiliary ADC input 1
30	X	RSVD	Reserved, tie to ground, either directly or through a pull-down resistor.
31	D_OUT	OSCO	OSCO is the output of the crystal oscillator circuit. A 100 Ω damping resistor should be connected between this pin and the crystal. This output should not be used to directly drive a clock to another IC. If the crystal oscillator is not used, this pin can be left unconnected.
32	D_IN	MCLKI	MCLKI can either be connected to a 3.3 V clock signal or can be the input from the crystal oscillator circuit.
33	PWR	PGND	PGND is the PLL ground pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. PGND should be decoupled to

			PVDD with a 100 nF capacitor.
34	PWR	PVDD	PVDD is the 3.3 V power supply for the PLL and the auxiliary ADC's analog section. This should be decoupled to PGND with a 100 nF capacitor.
35	A_OUT	PLL_LF	PLL Loop Filter connection. Two capacitors and a resistor need to be connected to this pin as shown in the Setting Master Clock/PLL Mode section.
36	PWR	AVDD	AVDD is a 3.3 V Analog Supply. This should be decoupled to AGND with a 100 nF capacitor.
37	PWR	AGND	AGND is an analog ground pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. AGND should be decoupled to an AVDD pin with a 100 nF capacitor.
38	D_IN	PLL_MODE0	PLL_MODE0 and PLL_MODE1 set the output frequency of the master clock PLL. See the Setting Master Clock/PLL Mode section for more details.
39	D_IN	PLL_MODE1	
40	A_OUT	CM	CM is the common mode reference. A 47 µF decoupling capacitor should be connected between this pin and ground to reduce crosstalk between the ADCs and DACs. The capacitor's material is not critical.
41	A_OUT	FILTD	FILTD is the DAC filter decoupling pin, which should be connected to a 10 µF capacitor to ground. The capacitor's material is not critical.
42	PWR	AGND	AGND is an analog ground pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. AGND should be decoupled to an AVDD pin with a 100 nF capacitor.
43	A_OUT	VOUT3	VOUT0-3 are the DAC outputs. Full-scale output voltage is 0.9 Vrms. These outputs can be used with either active or passive output reconstruction filters.
44	A_OUT	VOUT2	
45	A_OUT	VOUT1	
46	A_OUT	VOUT0	
47	A_OUT	FILTA	FILTA is the ADC filter decoupling pin, which should be connected to a 10 µF capacitor to ground. The capacitor's material is not critical.
48	PWR	AVDD	AVDD is a 3.3 V Analog Supply. This should be decoupled to AGND with a 100 nF capacitor.

[1] PWR = Power/Ground, A_IN = analog input, D_IN = digital input, A_OUT = analog output, D_IO = digital input/output, D/A_IO = digital/analog input/output

OVERVIEW

The core of the ADAU1701 is a 28-bit DSP (56-bit with double precision) optimized for audio processing. The program and parameter RAMs can be loaded with a custom audio processing signal flow built with ADI's SigmaStudio graphical programming software. The values stored in the parameter RAM control individual signal processing blocks, such as IIR equalization filters, dynamics processors, audio delays, and mixer levels. A safeload feature allows parameters to be transparently updated without causing clicks on the output signals.

The program RAM, parameter RAM, and register contents can be saved in an external EEPROM, from which the ADAU1701 can self-boot on start-up. In this stand-alone mode, parameters can be controlled through the on-board multipurpose pins. The ADAU1701 can accept controls from switches, potentiometers, rotary encoders, and IR receivers. Parameters such as volume and tone settings can be saved to the EEPROM on power-down and recalled when it is powered up again.

The ADAU1701 can operate with either digital I/Os, analog I/Os, or a mix of both. The stereo ADC and four-channel DAC each have an SNR of at least 100 dB and THD+N of at least -80 dB. The flexible serial data input/output ports allow for glueless interconnection to a variety of ADCs, DACs, general-purpose DSPs, S/PDIF receivers & transmitters, and sample rate converters. The ADAU1701 can be configured in I^2S, left-justified, right-justified, or TDM serial port compatible modes.

Twelve multi-purpose (MP) pins allow for the ADAU1701 to input external control signals and output flags or controls to other devices in the system. These MP pins can be configured as digital I/Os, inputs to the 4-channel auxiliary ADC, or set up as the serial data I/O ports. As inputs, they can be connected to buttons, switches, rotary encoders, potentiometers, IR receivers, or other external control circuitry to control the internal signal processing program. When configured as outputs, these pins can be used to drive LEDs (with a buffer), control other ICs, or connect to other external circuitry in an application.

The ADAU1701 has a sophisticated control port that supports complete read/write capability of all memory locations. Control registers are provided to offer complete control of the chip's configuration and serial modes. Handshaking is included for ease of memory uploads and downloads. The ADAU1701 can be configured for either SPI or I^2C control.

An on-board oscillator can be connected to an external crystal to generate the master clock. Also, a master clock phase-locked loop (PLL) allows the ADAU1701 to be clocked from a variety of different clock speeds. The PLL can accept inputs of $64 \times f_S$, $256 \times f_S$, $384 \times f_S$, or $512 \times f_S$ to generate the core's internal master clock.

The SigmaStudio software is used to program and control the SigmaDSP through the control port. Along with designing and tuning a signal flow, the tools can configure the all registers and burn a new program into the external EEPROM. SigmaStudio's graphical interface allows anyone with digital or analog audio processing knowledge to easily design a DSP signal flow and port it to a target application. It also provides enough flexibility and programmability for an experienced DSP programmer to have in-depth control of the design. In SigmaStudio, the user can simply connect graphical blocks such as biquad filters, dynamics processors, mixers, and delays, compile the design, and load the program and parameter files into the ADAU1701's memory through the control port. Signal processing blocks available in the provided libraries include

- Single- and double-precision biquad filters
- Mono and multichannel dynamics processors with peak or RMS detection
- Mixers and splitters
- Tone and noise generators
- Fixed and variable gain
- Loudness
- Delay
- Stereo enhancement
- Dynamic bass boost
- Noise and tone sources
- Level detectors
- GPIO control & conditioning

More processing blocks are always in development. Analog Devices also provides proprietary and third-party algorithms for applications such as matrix decoding, bass enhancement, and surround virtualizers. Please contact ADI for information about licensing these algorithms.

The ADAU1701 operates from a 1.8 V digital power supply and a 3.3 V analog supply. An on-board voltage regulator can be used to operate the chip from a single 3.3 V supply. It is fabricated on a single monolithic integrated circuit and is housed in a 48-lead LQFP package for operation over the –0°C to +70°C temperature range.

INITIALIZATION

POWER-UP SEQUENCE

The ADAU1701 has a built-in power-up sequence that initializes the contents of all internal RAMs on power-up or when the part is brought out of reset. After RESETB (positive edge-triggered) goes high, the contents of the internal program boot ROM are copied to the internal program RAM memory, the parameter RAM (all zeros) is filled with values from its associated boot ROM, and all registers are initialized to all-zeros. The default boot ROM program simply copies inputs to outputs with no processing (Figure 15). In this program, serial digital inputs 0-1 are output on DACs 0-1 and serial digital outputs 0-1. ADCs 0-1 are output on DACs 2-3. The data memories are also zeroed at power-up. New values should not be written to the control port until the initialization is complete.

Table 12. Power-up time

MCLKI input	Init. time	Maximum Program/ Parameter/ Register Boot Time (I^2C)	Total
3.072 MHz (64 × fs)	85 ms	175 ms	260 ms
11.289 MHz (256 × fs)	23 ms	175 ms	198 ms
12.288 MHz (256 × fs)	21 ms	175 ms	196 ms
18.432 MHz (384 × fs)	16 ms	175 ms	191 ms
24.576 MHz (512 × fs)	11 ms	175 ms	186 ms

The PLL start-up time lasts for 2^{18} cycles of the clock on the MCLKI pin. This time will range from 10.7 ms for a 24.576 MHz (512 × fs) input clock to 85.3 ms for a 3.072 MHz (64 × fs) input clock. This start-up time is measured from the rising edge of RESETB. Following the PLL start-up the ADAU1701's boot cycle takes 2048 cycles of the internal master clock (49.152 MHz at f_s = 48 kHz). The user should avoid writing to or reading from the ADAU1701 during this start-up time. For a 12.288 MHz input MCLK, the full initialization sequence (PLL start-up plus boot cycle) will last approximately 22 ms. Coming out of reset, the clock mode is immediately set by the PLL_MODE0 and PLL_MODE1 pins. Reset is synched to the falling edge of the internal MCLK.

Table 12 shows examples of typical times to boot the ADAU1701 into an application's operational state, assuming a 400 kHz I^2C clock loading a full program, parameter set, and all registers (8.5 kB). In reality, most applications will use less than this full amount, so the boot time (column 3) will be shorter.

RECOMMENDED PROGRAM/PARAMETER LOADING PROCEDURE

When writing large amounts of data to the program or parameter RAM in direct write mode, the processor core should be disabled to prevent unpleasant noises from appearing at the audio output.

1. Assert bits 3 and 4 (active low) of the core control register to mute the ADCs and DACs. This begins a volume ramp-down.
2. Assert bit 2 (active low) of the core control register. This zeroes the SigmaDSP's accumulators, the data output registers, and the data input registers.
3. Fill the program RAM using burst-mode writes.
4. Fill the parameter RAM using burst-mode writes.
5. Deassert bits 2-4 of the core control register.

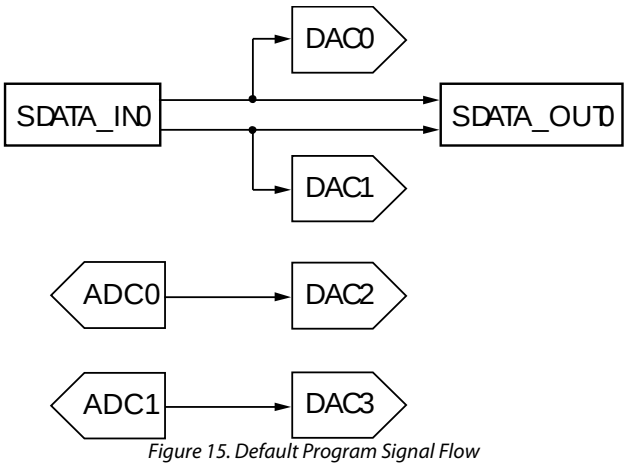


Figure 15. Default Program Signal Flow

POWER REDUCTION MODES

Sections of the ADAU1701 chip can be turned on and off as needed to reduce power consumption. These include the ADCs, DACs, and voltage reference.

The individual analog sections can be turned off in the Auxiliary ADC & Power Control Register (2082). By default, the ADCs, all four DACs, and reference are enabled (all bits set to 0). Each of these can be turned off by writing a 1 to the appropriate bits in this register. The ADC power-down mode will power down both ADCs and each DAC can be powered down individually. The current savings will be about 15 mA when the ADCs are powered down, and about 4 mA for each powered-down DAC. The voltage reference, which is supplied to both the ADCs and DACs, should only be powered down if the ADCs and all four DACs are also powered down. The reference is powered down by setting both bits 6 and 7 of the control register.

USING THE OSCILLATOR

The ADAU1701 has an on-board oscillator that can be used to generate the part's master clock. The oscillator is designed to work with a 256 × f_s master clock, which will be 12.288 MHz for

f_s = 48 kHz and 11.2896 MHz for f_s = 44.1 kHz. The crystal in the oscillator circuit should be an AT-cut parallel resonance device operating at its fundamental frequency. Figure 16 shows the recommended external circuit for proper operation.

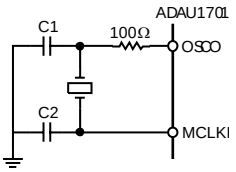


Figure 16. Crystal oscillator circuit

The 100 Ω damping resistor on OSCO will give the oscillator a voltage swing of approximately 2.2 V. The crystal's shunt capacitance should be 7 pF. Its load capacitance should be about 18 pF, although the circuit will support values up to 25 pF. The necessary values of the load capacitors C1 and C2 can be calculated from the crystal's load capacitance with the equation:

$$C_L = \frac{C_1 \times C_2}{C_1 + C_2} + C_{stray}$$

C_{stray} is the stray capacitance in the circuit and can usually be assumed to be 2-5 pF.

OSCO should not be used to directly drive the crystal's signal to another IC.

If the oscillator is not being used in the design it can be powered down to save system power. This would be done in a case where a system master clock is already available in the system. By default, the oscillator is powered on. The oscillator will power down when a 1 is written to bit 2 of the Oscillator Power-down Register (2086), as shown in Table 13.

Table 13. Oscillator Power-down Register (2086)

Register Bits	Function
15:3	Reserved, set to 0
2	Oscillator power-down
1:0	Reserved, set to 0

SETTING MASTER CLOCK/PLL MODE

The ADAU1701's MCLK input feeds a PLL, which generates the 1024 × f_s clock (49.152 MHz at f_s = 48 kHz) to run the DSP core. In normal operation, the input to MCLK must be one of the following: 64 × f_s, 256 × f_s, 384 × f_s, or 512 × f_s, where f_s is the input sampling rate. The mode is set on PLL_MODE0, and PLL_MODE1, according to Table 14. If the ADAU1701 is set to receive double-rate signals (by reducing the number of program steps/sample by a factor of 2 using the core control register), then the master clock frequencies must be either 32 × f_s, 128 × f_s, 192 × f_s, or 256 × f_s. If the ADAU1701 is set to receive quad-rate signals (by reducing the number of program steps/sample by a factor of 4 using the core control register), then the master clock frequencies must be one of 16 × f_s, 64 × f_s, 96 × f_s, or 128 × f_s. On power-up, a clock signal must be present on MCLK so that the ADAU1701 can complete its initialization routine.

Table 14. PLL Modes

MCLKI Input	PLL_MODE0	PLL_MODE1
64 × f_S	0	0
256 × f_S	0	1
384 × f_S	1	0
512 × f_S	1	1

The clock mode should not be changed without also resetting the ADAU1701. If the mode is changed on the fly, a click or pop may result on the outputs. The state of the PLL_MODEx pins should be changed while RESETB is held low.

The PLL's loop filter should be connected to the PLL_LF pin. This filter, shown in Figure 17, includes three passive components – two capacitors and a resistor. The values of these components does not need to be exact; the tolerance can be up to 10% for the resistor and 20% for the capacitors. The 3.3 V signal shown in the schematic can be connected to the chip's AVDD supply.

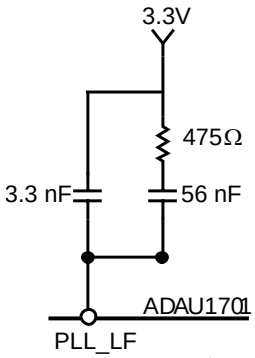


Figure 17. PLL Loop Filter

VOLTAGE REGULATOR

The ADAU1701's digital voltage must be set to 1.8 V. The chip includes an on-board voltage regulator that allows it to be used in systems where a 1.8 V supply is not available, but 3.3 V is. The only external components needed for this are a PNP transistor, one resistor, and bypass capacitors. Only one pin, VDRIVE, is necessary to support the regulator.

The recommended design for the voltage regulator is shown in Figure 18. The 10 µF and 100 nF capacitors shown in this schematic are recommended for bypassing, but are not necessary for operation. Each DVDD pin should have its own 100 nF bypassing capacitor, but only one bulk capacitor (10 µF) is needed for all pins. Here, 3.3 V is the main system voltage. 1.8 V is generated at the transistor's collector, which is connected to the DVDD pins. VDRIVE is connected to the base of the PNP transistor. If the regulator is not used in the design VDRIVE can be tied to ground.

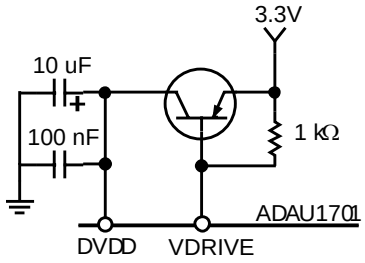


Figure 18. Voltage Regulator Design

Two specifications need to be taken into consideration when choosing a regulator transistor. First, the transistor's current amplification factor (h_{FE} or beta) should be at least 100. Second, the transistor's collector needs to be able to dissipate the heat generated when regulating from 3.3 V to 1.8 V. The maximum digital current draw of the ADAU1701 is 60 mA. The equation to determine the transistor's minimum power dissipation specs is as follows:

$$(3.3\ V - 1.8\ V) \times 60\ mA = 90\ mW$$

There are many transistors with these specifications available in small packages (i.e. SOT-23 or SOT-223), such as the 2N3906 or FZT953.

TWO-CHANNEL AUDIO ADC

The ADAU1701 has a two-channel Σ-Δ ADC. The SNR of the ADCs is 100 dB and the THD+N is -80 dB.

The stereo audio ADCs are current-input, so a voltage-to-current resistor is required on the inputs. This means that the voltage level of the input signals to the system can be set to any level; only the input resistors need to scale to provide the proper full-scale current input. The input pins ADC0 and ADC1, as well as ADC_RES have an internal 2 kΩ resistor for ESD performance. The external resistor connected to ADC_RES sets the full-scale current input of the ADCs. The full range of the ADC inputs is 100 µArms and this setting is given with an external 18 kΩ resistor (20 kΩ total, because it is in series with the internal 2 kΩ). The voltage seen directly on the ADC input pins will be the 1.5 V common mode.

The voltage-to-current resistors connected to ADC0/1 set the full-scale voltage input to the ADCs. With a full-scale current input of 100 µArms, a 2.0 Vrms signal with an external 18 kΩ resistor (in series with the 2 kΩ internal resistor) will give an input using the full range of the ADC. There should not be any need in an application to reduce the ADC's full-scale input by increasing the value of the resistor on ADC_RES.

Either input pins ADC0 and/or ADC1 can be left unconnected if that channel of the ADC is unused.

These calculations of resistor values all assume a 48 kHz sample rate. The recommended input and current setting resistors will scale directly with the sample rate because the ADCs have a switched-capacitor input. The total value (2 kΩ internal plus external resistor) of the ADC_RES resistor with sample rate f_{s_new} can be calculated by:

$$R_{total} = 20k\Omega \times \frac{48000}{f_{s\_new}}$$

The values of the resistors on the ADCx inputs can be calculated by:

$$R_{input\_total} = (rms\_input\_voltage) \times 10k\Omega \times \frac{48000}{f_{s\_new}}$$

Figure 19 shows a typical configuration of the ADC inputs for a 2.0 Vrms input signal. The 47 µF capacitors are used to ac-couple the signals so that the inputs are biased at 1.5 V.

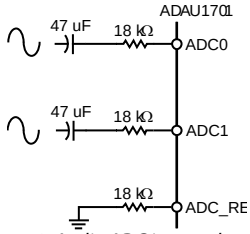


Figure 19. Audio ADC input schematic

FOUR-CHANNEL AUDIO DAC

The ADAU1701's main output is a four-channel Σ-Δ DAC. The SNR of the DAC is 104 dB and the THD+N is -90 dB. A full-scale output on the DACs is 0.9 Vrms (2.5 Vp-p).

The DACs are in an inverting configuration. If a signal inversion from input to output is undesirable, this can be reversed by using either an inverting configuration in the output filter, or by simply inverting the signal in the SigmaDSP program flow.

The DAC outputs can be filtered with either an active or a passive reconstruction filter. A single-pole low-pass filter with a 50 kHz corner frequency, as shown in Figure 20, is sufficient to filter the DAC's out-of-band noise, although an active filter may provide better audio performance. Figure 21 shows a 3-pole active low-pass filter that will provide a steeper roll-off and better stop-band attenuation than the passive filter.

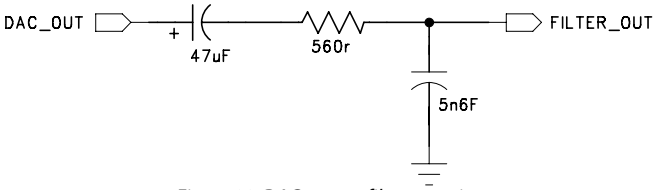


Figure 20. DAC output filter - passive

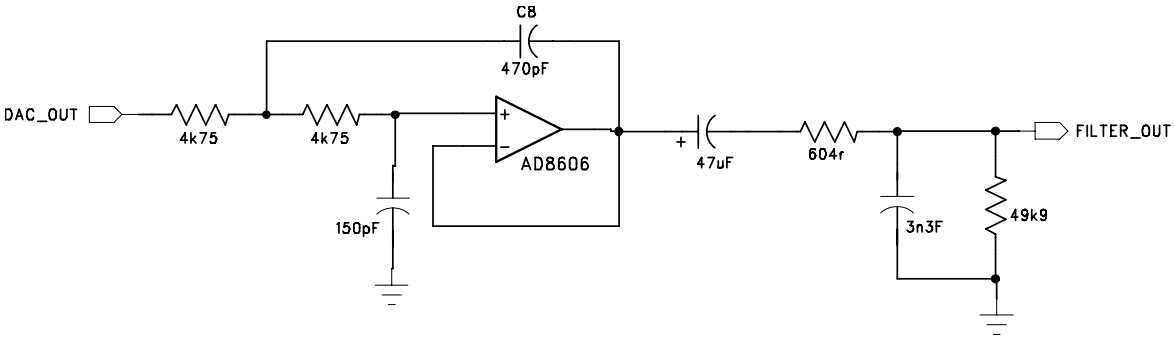


Figure 21. DAC output filter – active

CONTROL PORT

OVERVIEW

The ADAU1701 can operate in one of three control modes:

- I²C Control
- SPI Control
- Selfboot (no external controller)

The ADAU1701 has both a 4-wire SPI control port, and a 2-wire I²C bus control port that can each be used to set the part's RAMs and registers. When selfboot mode is not selected at power-up, the part defaults to I²C mode, but can be put into SPI control mode by pulling pin CLATCH/WP low three times. When the SELFBOOT pin is set high, the ADAU1701 will load its program, parameters, and registers settings from an external EEPROM on start-up.

The control port is capable of full read/write operation for all of the memories and registers. Most signal processing parameters are controlled by writing new values to the parameter RAM using the control port. Other functions, such as mute and input/output mode control, are programmed by writing to the control registers.

All addresses may be accessed in both a single-address mode or a burst mode. A control word consists of the chip address, the register/RAM subaddress, and the data to be written. The number of bytes per word depends on the type of data that is written.

The first byte of a control word (Byte 0) contains the 7-bit chip address plus the R/W bit. The next two bytes (Bytes 1 and 2) together form the subaddress of the memory or register location within the ADAU1701. This subaddress needs to be two bytes because the memories within the ADAU1701 are directly addressable, and their sizes exceed the range of single-byte addressing. All subsequent bytes (Bytes 3, 4, etc.) contain the data, such as control port data or program or parameter data. The exact formats for specific types of writes are shown in Table 27 to Table 35.

The ADAU1701 has several mechanisms for updating signal processing parameters in real time without causing pops or clicks. In cases where large blocks of data need to be down-loaded, the output of the DSP core can be halted (using Bit x of the core control register), new data loaded, and then restarted. This is typically done during the booting sequence at start-up or when loading a new program into RAM. In cases where only a few parameters need to be changed, they can be loaded without halting the program. To avoid unwanted side effects while loading parameters on the fly, the SigmaDSP provides the safeload registers. The safeload registers can be used to buffer a full set of parameters (e.g. the five coefficients of a biquad) and then transfer these parameters into the active program within one audio frame. The safeload mode uses internal logic to prevent contention between the DSP core and the control port.

The control port pins are multi-functional according to which mode in which the part is operating. details these different functions.

I²C PORT

The ADAU1701 supports a 2-wire serial (I²C compatible) microprocessor bus driving multiple peripherals. Two pins, serial data (SDA) and serial clock (SCL), carry information between the ADAU1701 and the system I²C master controller. In I²C mode the ADAU1701 is always a slave on the bus, which means that it will never initiate a data transfer. Each slave device is recognized by a unique address. The address byte format is shown in Table 15. The ADAU1701 has four possible slave addresses: two for writing operations and two for reading. These are unique addresses for the device and are illustrated in Table 16. The LSB of the byte sets either a read or write operation; Logic Level 1 corresponds to a read operation, and Logic Level 0 corresponds to a write operation. The sixth and seventh bits of the address are set by tying the ADDRx pins of the ADAU1701 to logic level 0 or logic level 1. Both SDA and SCL should have 2.2 kΩ pull-up resistors on the lines connected to them. The voltage on these signal lines should not be above IOVDD (3.3 V).

Table 15. ADAU1701 Address Byte Format

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

Table 16. ADAU1701 I²C Addresses

ADDR1	ADDR0	Read/$\overline{Write}$	Slave Address
0	0	0	0x68
0	0	1	0x69
0	1	0	0x6A
0	1	1	0x6B
1	0	0	0x6C
1	0	1	0x6D
1	1	0	0x6E
1	1	1	0x6F

Addressing

Initially, all devices on the I²C bus are in an idle state, which is where the devices monitor 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. This indicates that an address/data stream will follow. All devices on the bus respond to the start condition and shift the next eight bits (7-bit address + 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 means the master will write information to the peripheral. A logic 1 on the LSB of the first byte means the master will read information from the peripheral. 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. Figure 22 shows the timing of an I²C write.

Burst mode addressing, where the subaddresses are automatically incremented at word boundaries, can be used for writing large amounts of data to contiguous memory locations. This increment happens automatically if a stop condition is not encountered after a single-word write. The registers and memories in the ADAU1701 range in width from one to five bytes, so the autoincrement feature knows the mapping between sub-addresses and the word length of the destination register (or memory location). A data transfer is always terminated by a stop condition.

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, these cause an immediate jump to the idle condition. During a given SCL high period, the user should only issue one start condition, one stop condition, or a single stop condition followed by a single start condition. If an invalid subaddress is issued by the user, the ADAU1701 does not issue an acknowledge and returns to the idle condition. If the user exceeds the highest subaddress while in autoincrement mode, one of two actions will be taken. In read mode, the ADAU1701 outputs the highest subaddress register contents until the master device issues a no-acknowledge, indicating the end of a read. A no-acknowledge condition is where the SDA line is not pulled low on the ninth clock pulse on SCL. If the highest subaddress location is reached while in write mode, the data for the invalid byte is not loaded into any subaddress register, a no-acknowledge is issued by the ADAU1701, and the part returns to the idle condition.

I²C Read & Write Operations

Figure 24 shows the timing of a single-word write operation. Every ninth clock, the ADAU1701 issues an acknowledge by pulling SDA low.

Figure 25 shows the timing of a burst mode write sequence. This figure shows an example where the target destination registers are two bytes. The ADAU1701 knows to increment its subaddress register every two bytes because the requested subaddress corresponds to a register or memory area with a 2-byte word length.

The timing of a single word read operation is shown in Figure 26. Note that the first R/$\overline{W}$ bit is still a 0, indicating a write operation. This is because the subaddress still needs to be written in order to set up the internal address. After the ADAU1701 acknowledges the receipt of the subaddress, 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 ADAU1701's SDA to turn around and begin driving data back to the master. The master then responds every ninth pulse with an acknowledge pulse to the ADAU1701.

Figure 27 shows the timing of a burst-mode read sequence. This figure shows an example where the target read registers are two bytes. The ADAU1701 knows to increment its subaddress register every two bytes because the requested subaddress corresponds to a register or memory area with word lengths of two bytes. Other address ranges may have a variety of word lengths ranging from one to five bytes; the ADAU1701 always decodes the subaddress and sets the autoincrement circuit so that the address increments after the appropriate number of bytes.

SPI PORT

By default, the ADAU1701 is in I²C mode, but can be put into SPI control mode by pulling CLATCH/WP low three times. The SPI port uses a 4-wire interface, consisting of CLATCH, CCLK, CDATA, and COUT signals. The CLATCH signal goes low at the beginning of a transaction and high at the end of a transaction. The CCLK signal latches CDATA on a low-to-high transition. COUT data is shifted out of the ADAU1701 on the falling edge of CCLK and should be clocked into the receiving device, such as a microcontroller, on CCLK's rising edge. The CDATA signal carries the serial input data, and the COUT signal is the serial output data. The COUT signal remains three-stated until a read operation is requested. This allows other SPI-compatible peripherals to share the same readback line. All SPI transactions follow the same basic format, shown in Table 17. A timing diagram is shown in Figure 39. All data written should be MSB-first.

Table 17. Generic Control Word Format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4, etc.
chip_adr [6:0], R/$\overline{W}$	0000, subadr [11:8]	subadr[7:0]	data	data

Chip Address R/$\overline{W}$

The first byte of an SPI transaction includes the 7-bit chip address and a R/$\overline{W}$ bit. The chip address is set by the ADR_SEL pin. This allows two ADAU1701s to share a CLATCH signal, yet still operate independently. When ADR_SEL0 is low, the chip address is 0000000; when it is high, the address is 0000001. The LSB of this first byte determines whether the SPI transaction is a read (Logic Level 1) or a write (Logic Level 0).

Subaddress

The 12-bit Subaddress word is decoded into a location in one of the memories or registers. This subaddress is the location of the appropriate RAM location or register.

Data Bytes

The number of data bytes varies according to the register or memory being accessed. In burst write mode, an initial subaddress is given followed by a continuous sequence of data for consecutive memory/register locations. The detailed data format diagram for continuous-mode operation is given in the Control Port Read/Write Data Formats section.

A sample timing diagram for a single SPI write operation to the parameter RAM is shown in Figure 28. A sample timing diagram of a single SPI read operation is shown in Figure 29. The COUT pin goes from three-state to driven at the beginning of Byte 3. In this example, Bytes 0 to 2 contain the addresses and R/W bit, and subsequent bytes carry the data.

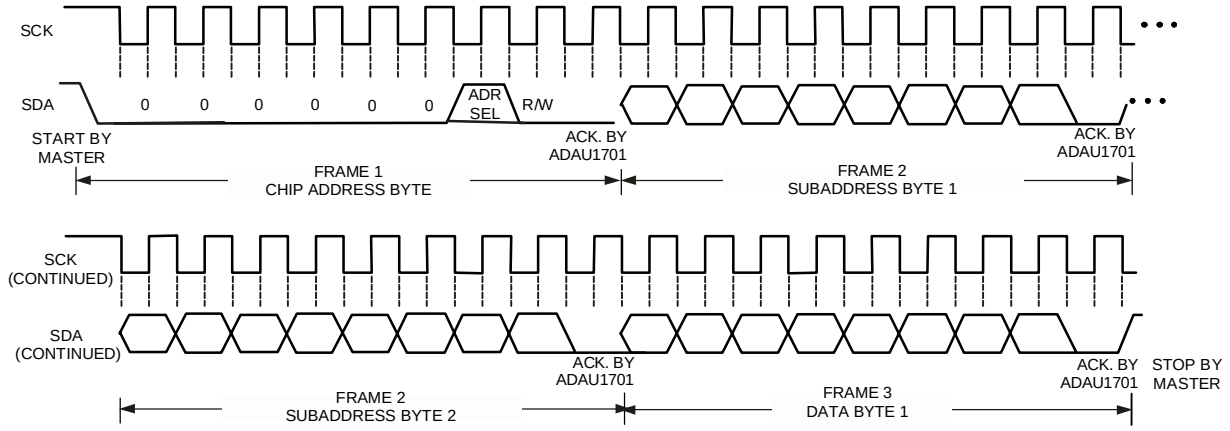


Figure 22. ADAU1701 I2C Write Clocking

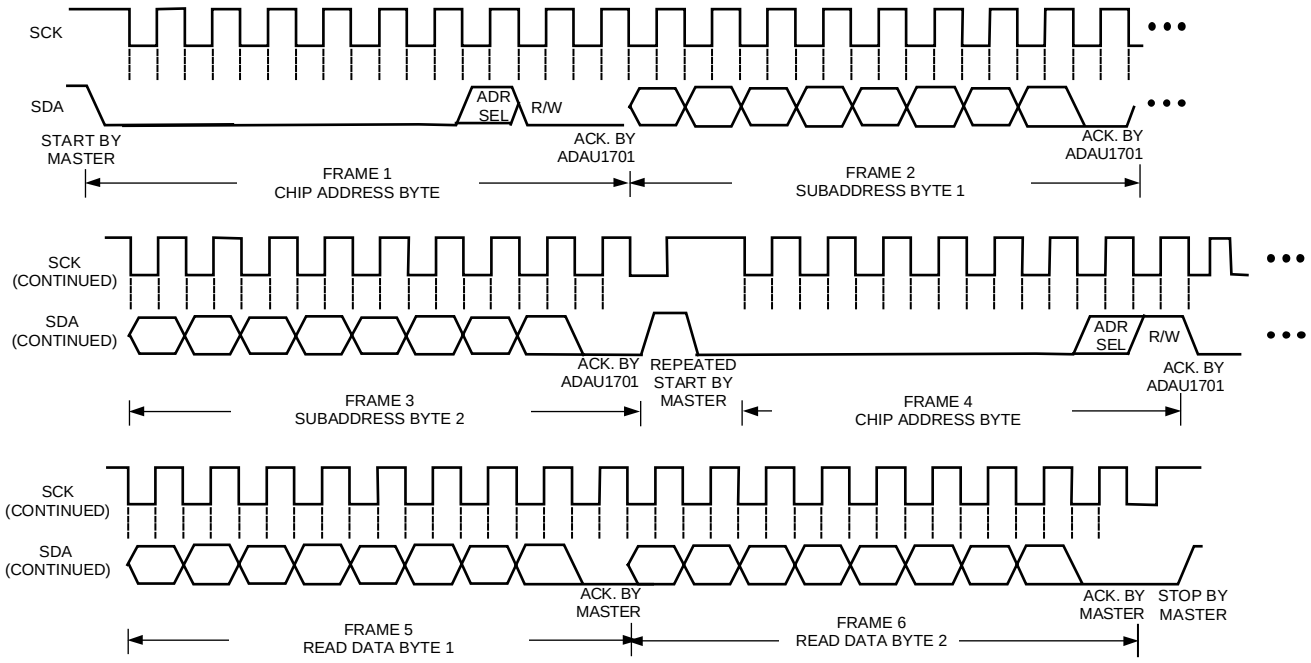


Figure 23. ADAU1701 I²C Read Clocking

S	Chip Address, R/$\overline{W}$ = 0	AS	Subaddress High	AS	Subaddress Low	AS	Data Byte 1	AS	Data Byte 2	…	AS	Data Byte N	P

Figure 24. Single-Word I²C Write Format

S	Chip Address, R/$\overline{W}$ = 0	AS	Subaddress High	AS	Subaddress Low	AS	Data Word 1, Byte 1	AS	Data Word 1, Byte 2	AS	Data Word 2, Byte 1	AS	Data Word 2, Byte 2	AS	…	P

Figure 25. Burst Mode I²C Write Format

S	Chip Address, R/$\overline{W}$ = 0	AS	Subaddress High	AS	Subaddress Low	AS	S	Chip Address, R/$\overline{W}$ = 1	AS	Data Byte 1	AM	Data Byte 2	…	AM	Data Byte N	P

Figure 26. Single Word I²C Read Format

S	Chip Address, R/$\overline{W}$ = 0	AS	Subaddress High	AS	Subaddress Low	AS	S	Chip Address, R/$\overline{W}$ = 1	AS	Data Word 1, Byte 1	AM	Data Word 1, Byte 2	AM	…	P

Figure 27. Burst Mode I²C Read Format

S - Start Bit
P - Stop Bit
AM - Acknowledge by Master
AS - Acknowledge by Slave

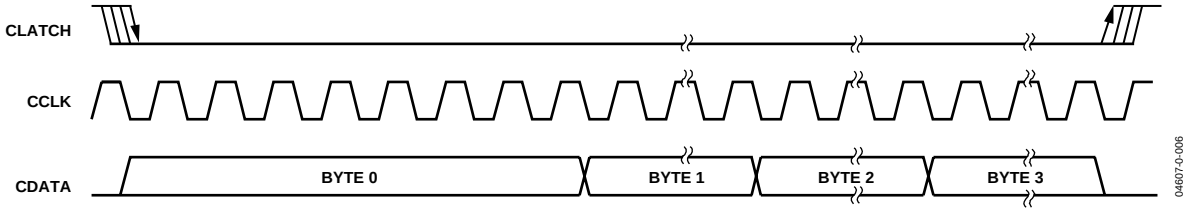


Figure 28. SPI Write Clocking (Single-Write Mode)

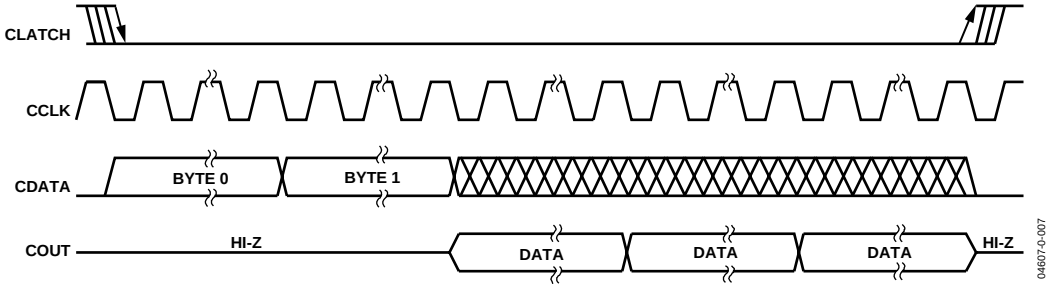


Figure 29. SPI Read Clocking (Single-Read Mode)

Table 18. Control Port/Selfboot Pin Functions

Pin	I²C Mode	SPI Mode	Selfboot
SCL/CCLK	SCL - input	CCLK - input	SCL - output
SDA/COUT	SDA – open collector output	COUT – output	SDA – open collector output
ADDR1/CDATA/WB	ADDR1 - input	CDATA - input	Writeback trigger
CLATCH/WP	Unused input – tie to ground or VDD	CLATCH - input	EEPROM Write Protect - open collector output
ADDR0	ADDR0 - input	ADDR0 - input	unused input – tie to ground or VDD

SELF BOOT

On power-up, the ADAU1701 can load a set of program and parameters that are saved in an external EEPROM. Combined with the auxiliary ADC and the multipurpose pins, this eliminates the need for a microcontroller in the system. The self-booting is accomplished by the ADAU1701 acting as a master on the I²C bus on start-up, which occurs when the Selfboot pin is set high. The ADAU1701 cannot self-boot in SPI mode.

The maximum necessary EEPROM size is 9,248 bytes, or just over 9 kB. This much memory will only be needed if the program RAM (1024 × 5 bytes), parameter RAM (1024 × 4 bytes), and interface registers (8 × 4 bytes) are each completely full. In most applications, an 8 kB EEPROM will be sufficient.

A selfboot operation is triggered on the rising edge of RESETB when the SELFBOOT and WP pins are set high. The ADAU1701 reads a program, parameters, and register settings from the EEPROM. Once the ADAU1701 has finished selfbooting, further messages may be sent to the ADAU1701 on the I²C bus, although this typically won't be necessary in a selfbooting application. The I²C device address is 0x68 for a write and 0x69 for a read in this mode. The ADDRx pins have different functions when the chip is in this mode, so the settings on them are ignored.

The ADAU1701 will not selfboot if WP is set low. Holding this pin low allows the EEPROM to be programmed in-circuit. The WP pin must be pulled low (it will normally have a resistor pull-up) to enable writes to the EEPROM and this disables selfboot until WP is taken high.

The ADAU1701 is a master on the I²C bus during selfboot and writeback. While it would be uncommon for an application using selfboot to also have a microcontroller in on the control lines, care should be taken that no other device tries to write to the I²C bus during selfboot or writeback. The ADAU1701 generates SCL at 8 × fs, so at fs = 48 kHz SCL will run at 384 kHz. SCL has a duty cycle of 3/8 in accordance with the I²C specification.

The ADAU1701 reads from EEPROM chip address 0x61. The LSBs of the addresses of some EEPROMs are pin-configurable; in most cases these pins should be tied low to set this address shown here.

EEPROM format

The EEPROM data contains a sequence of messages. Each discrete message will be one of the seven types defined in Table 19. Each message consists of a sequence of one or more bytes. The first byte identifies the message type. Bytes are written MSB-first. Most messages will be block write (0x01) types, which are used for writing to the ADAU1701's program RAM, parameter RAM, and control registers.

The body of the message following the message type should start with a 0x00 byte – this is the chip address. After this there is always a 2-byte register/memory address field, as there are with all other control port transactions.

WriteBack

A writeback occurs when the WB pin is triggered and data is written to the EEPROM from the ADAU1701. This function is typically used to save volume and other parameter settings to the EEPROM just before power is removed from the system. A rising edge on the WB pin triggers a writeback when the ADAU1701 is in selfboot mode, unless a Set WB to Falling Edge Sensitive (0x05) message was contained in the selfboot message sequence. Only one write back will take place unless a Set Multiple Write Back (0x04) message was contained in the selfboot message sequence. The WP pin will be pulled low when a writeback is triggered in order to allow writing to the EEPROM.

The ADAU1701 is only capable of writing back the contents of the interface registers to the EEPROM. These registers are usually set by the DSP program, but can also be written to directly after setting bit 6 of the

Table 19. EEPROM Message Types

Message ID	Message Type	Following Bytes
0x00	End	none
0x01	Write	2 bytes indicating message length followed by appropriate number of data bytes
0x02	Delay	2 bytes for delay
0x03	No-Op	none
0x04	Set multiple write back	none
0x05	Set WB to falling edge sensitive	none
0x06	End and wait for writeback	none

DSP Core Control Register. The parameter settings that should be saved are configured in SigmaStudio.

The writeback functions by writing data from the ADAU1701's interface registers to the second page of the selfboot EEPROM, addresses 32-63. Starting at EEPROM address 26 (so that the interface register data begins at address 32), the EEPROM should be programmed with six bytes - the Message Byte (0x01), 2 length bytes, the chip address (0x00), the 2-byte subaddress for the interface registers (0x08 0x00). There must be a message to the DSP core control register to enable port writing to the interface registers prior to the interface register data in the EEPROM. This should be stored in EEPROM address 0. No-op messages (0x03) may be used in-between messages to ensure that these conditions are met.

Table 20 shows an example of what should be stored in the EEPROM starting with EEPROM address 0. In this example, the interface registers are first set to control port write mode (line 1), which is followed by 18 no-op bytes (lines 2-4) so that the interface register data will appear on page 2 of the EEPROM. Next follows the write header (line 4) and then 32 bytes of interface register data (lines 5-8). Finally, the program RAM data, starting at ADAU1701 address 0x04 0x00 is written (lines 9-11). In this example, the program length is 70 words, or 350 bytes, so 332 more bytes will be included in the EEPROM but are not shown here.

The ADAU1701 writes to EEPROM chip address 0x60. The LSBs of the addresses of some EEPROMs are pin-configurable; in most cases these pins should be tied low to set the address shown here.

The maximum number of bytes that will be written back from the ADAU1701 is 35 (eight 4-byte Interface registers plus 3 bytes of EEPROM addressing overhead). With SCL running at 384 kHz, this means that the writeback operation will take approximately 73 µs to complete after being triggered. Care should be taken to ensure that sufficient power is available to the system for enough time to allow a writeback to complete, especially if the WB signal is triggered from a falling power supply voltage.

Table 20. EEPROM Writeback Example

0x01	0x00	0x05	0x00	0x08	0x1C	0x00	0x40
Write	Length		Device Addr.	Core Control Register address		Core Control Register Data	
0x03	0x03	0x03	0x03	0x03	0x03	0x03	0x03
No-Op bytes							
0x03	0x03	0x03	0x03	0x03	0x03	0x03	0x03
No-Op bytes							
0x03	0x03	0x01	0x00	0x23	0x00	0x08	0x00
No-Op bytes		Write	Length		Device Addr.	Interface Register Address	
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Interface Register Data							
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Interface Register Data							
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Interface Register Data							
0x00	0x00	0x00	0x00	0x00	0x00	0x00	0x00
Interface Register Data							
0x01	0x01	0x61	0x00	0x04	0x00	0x00	0x00
Write	Length		Device Addr.	Program RAM address		Program RAM data	
0x00	0x00	0x01	0x00	0x00	0x00	0xE8	0x01
Program RAM data							
0x00	0x00	0x00	0x00	0x01	0x00	0x08	0x00
Program RAM data (continues for 332 more bytes)							

SIGNAL PROCESSING

OVERVIEW

The ADAU1701 is designed to provide all signal processing functions commonly used in stereo or multichannel playback systems. The signal processing flow is designed using the ADI-supplied SigmaStudio software, which allows graphical entry and real-time control of all signal processing functions.

Many of the signal processing functions are coded using full, 56-bit double-precision arithmetic. The input and output word lengths are 24 bits. Four extra headroom bits are used in the processor to allow internal gains up to 24 dB without clipping. Additional gains can be achieved by initially scaling down the input signal in the signal flow.

The signal processing blocks can be arranged in a custom program that can be loaded to the ADAU1701's RAM. The available signal processing blocks are explained in the following sections.

NUMERIC FORMATS

It is common in DSP systems to use a standardized method of specifying numeric formats. Fractional number systems are specified by an A.B format, where A is the number of bits to the left of the decimal point and B is the number of bits to the right of the decimal point.

The ADAU1701 uses the same numeric format for both the coefficient values (stored in the parameter RAM) and the signal data values. The format is as follows:

Numerical Format: 5.23

Range: –16.0 to (+16.0 – 1 LSB)

Examples:
1000 0000 0000 0000 0000 0000 0000 = –16.0
1110 0000 0000 0000 0000 0000 0000 = –4.0
1111 1000 0000 0000 0000 0000 0000 = –1.0
1111 1110 0000 0000 0000 0000 0000 = –0.25
1111 1111 1111 1111 1111 1111 1111 = (1 LSB below 0.0)
0000 0000 0000 0000 0000 0000 0000 = 0.0
0000 0010 0000 0000 0000 0000 0000 = 0.25
0000 1000 0000 0000 0000 0000 0000 = 1.0
0010 0000 0000 0000 0000 0000 0000 = 4.0
0111 1111 1111 1111 1111 1111 1111 = (16.0 – 1 LSB).

The serial port accepts up to 24 bits on the input and is sign-extended to the full 28 bits of the core. This allows internal gains of up to 24 dB without encountering internal clipping.

A digital clipper circuit is used between the output of the DSP core and the outputs (see Figure 30). This clips the top four bits of the signal to produce a 24-bit output with a range of 1.0 (minus 1 LSB) to –1.0.

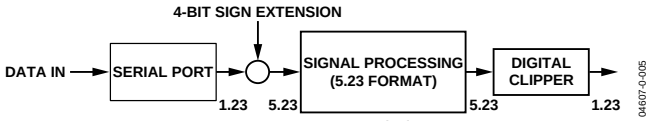


Figure 30. Numeric Precision and Clipping Structure

PROGRAMMING

On power-up, the ADAU1701's default program passes the unprocessed input signals to the outputs (Figure 15), but the outputs are muted by default (see Power-Up Sequence section). There are 1,024 instruction cycles per audio sample, resulting in an internal clock rate of 49.152 MHz (for f_s = 48 kHz). This DSP runs in a stream-oriented manner, meaning all 1,024 instructions are executed each sample period. The ADAU1701 may also be set up to accept double or quad-speed inputs by reducing the number of instructions/sample, which can be set in the core control register.

The part can be programmed easily using SigmaStudio, a graphical tool provided by Analog Devices. No knowledge of writing line-level DSP code is required.

RAMS AND REGISTERS

Table 21. Control Port Addresses

SPI/ I²C Subaddress	Register/RAM Name	Read/Write Word Length
0–1023 (0x0000–0x03FF)	Parameter RAM	Write: 4 Bytes, Read: 4 Bytes
1024–2047 (0x0400-0x07FF)	Program RAM	Write: 5 Bytes, Read: 5 Bytes
2048-2055 (0x0800-0x0807)	Interface Registers 0 - 7	Read: 4 bytes, Write: 4 bytes
2056 (0x0808)	GPIO Pin Setting Register	Read: 2 bytes, Write: 2 bytes
2057-2060 (0x0809-0x080C)	Aux ADC Data Registers	Read: 2 bytes, Write: 1 byte
2064–2068 (0x080D-0x0814)	Safeload Data Registers 0 – 4	Write: 5 Bytes, Read: N/A
2069–2073 (0x0815-0x0819)	Safeload Address Registers 0 - 4	Write: 2 Bytes, Read: N/A
2074-2075 (0x081A-0x081B)	Data Capture Registers 0–1	Write: 2 Bytes, Read: 3 Bytes
2076 (0x081C)	DSP Core Control Register	Write: 2 Bytes, Read: 2 Bytes
2077 (0x081D)	Reserved – do not write	Write: 1 Byte, Read: 1 Byte
2078 (0x081E)	Serial Output Control Register	Write: 2 Bytes, Read: 2 Bytes
2079 (0x081F)	Serial Input Control Register	Write: 1 Byte, Read: 1 Byte
2080-2081 (0x0820-0x0821)	Multi-Purpose Pin Configuration Registers 0 – 1	Write: 3 Bytes, Read: 3 Bytes
2082 (0x0822)	Auxiliary ADC & Power Control Register	Write: 2 Bytes, Read: 2 Bytes
2083 (0x0823)	Reserved – do not write	Write: 2 Bytes, Read: 2 Bytes
2084 (0x0824)	Auxiliary ADC Enable Register	Write: 2 Bytes, Read: 2 Bytes
2085 (0x0825)	Reserved – do not write	Write: 2 Bytes, Read: 2 Bytes
2086 (0x0826)	Oscillator Power-down Register	Write: 2 Bytes, Read: 2 Bytes

Table 22. RAM Read/Write Modes

Memory	Size	Address Range	Read	Write	Write Modes
Parameter RAM	1024 × 28	0–1023	Yes	Yes	Direct Write[1], Safeload Write
Program RAM	1024 × 40	1024–2047	Yes	Yes	Direct Write[1]

[1] Internal registers should be cleared first to avoid clicks/pops.

CONTROL PORT ADDRESSING

Table 21 shows the addressing of the ADAU1701's RAM and register spaces. The address space encompasses a set of registers and two RAMs: one each for holding signal processing parameters and holding the program instructions. The program and parameter RAMs are initialized on power-up from on-board boot ROMs (see Power-Up Sequence section).

Table 22 shows the sizes and available writing modes of the parameter and program RAMs.

All RAMs and registers have a default value of all zeros, except for the program RAM which is loaded with the default program as described in the Initialization section.

PARAMETER RAM

The parameter RAM is 28 bits wide and occupies Addresses 0 to 1023. The parameter RAM is initialized to all zeros on power-up. The data format of the parameter RAM is twos complement 5.23. This means that the coefficients may range from +16.0 (minus 1 LSB) to –16.0, with 1.0 represented by the binary word 0000 1000 0000 0000 0000 0000 0000 or hexadecimal word 0x00 0x80 0x00 0x00.

The parameter RAM can be written and read using one of the two following methods.

Direct Read/Write

This method allows direct access to the program and parameter RAMs. This mode of operation is normally used during a complete new load of the RAMs, using burst-mode addressing. The clear registers bit in the core control register should be set to 0 using this mode to avoid any clicks or pops in the outputs. Note that it is also possible to use this mode during live program execution, but since there is no handshaking between the core and the control port, the parameter RAM will be unavailable to the DSP core during control writes, resulting in clicks and pops in the audio stream.

Safeload Write

Up to five safeload registers can be loaded with parameter RAM address/data. The data is then transferred to the requested address when the RAM is not busy. This method can be used for dynamic updates while live program material is playing through the ADAU1701. For example, a complete update of one biquad section can occur in one audio frame, while the RAM is not busy. This method is not available for writing to the program RAM or control registers.

The following sections discuss these two options in more detail.

SAFELOAD REGISTERS

Many applications require real-time microcontroller control of signal processing parameters, such as filter coefficients, mixer gains, multi-channel virtualizing parameters, or dynamics processing curves. One example is that to prevent instability from occurring, all of the parameters of a biquad filter must be updated at the same time. Otherwise, the filter could execute for one or two audio frames with a mix of old and new coefficients. This mix could cause temporary instability, leading to transients that could take a long time to decay. To eliminate this problem, the ADAU1701 can simultaneously load a set of five 28-bit values to the desired parameter RAM address. Five registers are used because a biquad filter uses five coefficients, and it is desirable to be able to do a complete biquad update in one transaction.

The first step in performing a safeload is writing the parameter address to one of the Safeload Address Registers (2069 – 2073). The 10-bit data word that should be written is the address to which the safeload is being performed. After the Safeload Address Register is set, then the 28-bit data word can be written to the corresponding Safeload Data Register (2064 – 2068). The data formats for these writes are detailed in Table 35 and Table 36. Table 23 shows how each of the five Address Registers map to their corresponding Data Registers.

Table 23. Safeload Address & Data Register Mapping

Safeload Register	Safeload Address Register	Safeload Data Register
0	2069	2064
1	2070	2065
2	2071	2066
3	2072	2067
4	2073	2068

Once the address and data registers are loaded, the initiate safeload transfer bit in the core control register should be set to initiate the loading into RAM. Each safeload register will take one of the 1,024 core instructions to load into the parameter RAM. Total program lengths should be limited to 1,019 cycles (1,024 − 5) to ensure that the SigmaDSP core has at least five free cycles to perform the safeloads. It is guaranteed that the safeload will have occurred within one LRCLK period (21 µs at f_s = 48 kHz) of the initiate safeload transfer bit being set.

The safeload logic automatically sends only those safeload registers that have been written to since the last safeload operation. For example, if only two parameters are to be sent, only two of the five safeload registers must be written. When the initiate safeload transfer bit is asserted, only those two registers are sent; the other three registers are not sent to the RAM and can still hold old or invalid data.

DATA CAPTURE REGISTERS

The ADAU1701's data capture feature allows the data at any node in the signal processing flow to be sent to one of two control port-readable registers. This can be used to monitor and display information about internal signal levels or compressor/limiter activity.

For each of the data capture registers, a capture count and a register select must be set. The capture count is a number between 0 and 1023 that corresponds to the program step number where the capture will occur. The register select field programs one of four registers in the DSP core that will be transferred to the data capture register when the program counter equals the capture count. The register select field selections are shown in Table 24.

Table 24. Data Capture Control Registers (2074-2075)

Register Bits	Function
12:2	11-Bit Program Counter Address
1:0	Register Select

Table 25. Data Capture Output Register Select

Setting	Register
00	Multiplier X Input (Mult_X_input)
01	Multiplier Y Input (Mult_Y_input)
10	Multiplier-Accumulator Output (MAC_out)
11	Accumulator Feedback (Accum_fback)

The capture count and register select bits are set by writing to one of the eight data capture registers at register addresses

2074: Control Port Data Capture Setup Register 0
2075: Control Port Data Capture Setup Register 1

The captured data is in 5.19 two's complement data format. The four LSBs are truncated from the internal 5.23 data word.

The data that must be written to set up the data capture is a concatenation of the 11-bit program count index with the 2-bit register select field. The capture count and register select values that correspond to the desired point to be monitored in the signal processing flow can be found in a file output from the program compiler. The capture registers can be accessed by reading from locations 2074 and 2075. The format for reading and writing to the data capture registers can be seen in Table 33 and Table 34.

DSP CORE CONTROL REGISTER

The controls in this register set the operation of the ADAU1701's DSP core.

Table 26. DSP Core Control Register (2076)

Register Bits	Function
15:14	Reserved
13:12	GPIO Debounce control
	00 = 20ms
	01 = 40ms
	10 = 10ms
	11 = 5ms
11:9	Reserved
8	Aux ADC Data registers control port write mode
7	GPIO Pin Setting register control port write mode
6	Interface registers control port write mode
5	Initiate Safeload Transfer
4	Mute ADCs, active low
3	Mute DACs, active low
2	Clear Internal Registers to All Zeros, active low
1:0	Sample Rate
	00 = 1× (1024 instructions, 48 kHz)
	01 = 2× (512 instructions, 96 kHz)
	10 = 4× (256 instructions, 192 kHz)
	00 = reserved

GPIO Debounce control (Bits 13:12)

Set debounce time of multipurpose pins set as GPIO inputs.

Aux ADC Data registers control port write mode (Bit 8)

Setting this bit allows data to be written directly to the Aux ADC Data registers (2057-2060) from the control port. When set, the Aux ADC Data registers will no longer respond to settings on the multi-purpose pins.

GPIO Pin Setting register control port write mode (Bit 7)

When this bit is set, the GPIO Pin Setting register (2056) can be written to directly from the control port and this register will no longer respond to input settings on the multi-purpose pins.

Interface registers control port write mode (Bit 6)

When this bit is set, data can be written directly to the Interface registers (2048-2055) from the control port. In that state, the Interface registers will not be written from the SigmaDSP program.

Initiate Safe Transfer to Parameter RAM (Bit 5)

Setting this bit to 1 initiates a safeload transfer to the parameter RAM. This bit is automatically cleared when the operation is completed. There are five safeload register pairs (address/data); only those registers that have been written since the last safeload event are transferred to the parameter RAM.

Mute ADCs (Bit 4)

This bit will mute the output of the ADCs. The bit defaults to 0 and is active-low, so it must be set to 1 in order to pass audio from the ADCs.

Mute DACs (Bit 3)

This bit will mute the output of the DACs. The bit defaults to 0 and is active-low, so it must be set to 1 in order to pass audio from the DACs.

Clear Internal Registers to All Zeros (Bit 2)

This bit defaults to 0 and is active low. This bit needs to be set to 1 in order for a signal to pass through the SigmaDSP core.

Sample Rate (Bits 1:0)

These bits set the number of DSP instructions for every sample and the sample rate at which the ADAU1701 will operate. At the default setting of 1× there will be 1024 instructions per audio sample. This setting should be used with sample rates such as 48 kHz and 44.1 kHz.

In the 2× setting the number of instructions per frame will be halved to 512 and the ADCs and DACs will nominally run at a 96 kHz sample rate.

At a 4× setting there will be 256 instructions per cycle and the converters will run at a 192 kHz sample rate.

INTERFACE REGISTERS

The interface registers are used in self-boot mode to save parameters that need to be written to the external EEPROM. The ADAU1701 will then recall these parameters from the EEPROM after the next reset or power-up. This way, system parameters such as volume and EQ settings can be saved during power-down and recalled when the system is next turned on.

There are eight 32-bit interface registers, which allows for eight 28-bit (plus zero-padding) parameters to be saved. The parameters that will be saved in these registers are set in the graphical programming tools. These registers are updated with their corresponding parameter RAM data once per sample period.

An edge, which can be set to be either rising or falling, triggers the ADAU1701 to write the current contents of the interface registers to the EEPROM. See more information in the Self boot section.

The user can write directly to the interface registers after bit 6 in the DSP core control register has been set. In this mode, the data in the registers is written from the control port and not from the DSP core.

CONTROL PORT READ/WRITE DATA FORMATS

The read/write formats of the control port are designed to be byte-oriented. This allows for easy programming of common

microcontroller chips. In order to fit into a byte-oriented format, 0s are appended to the data fields before the MSB in order to extend the data word to the next multiple of eight bits. For example, 28-bit words written to the parameter RAM are appended with four leading 0s in order to reach 32 bits (4 bytes); 40-bit words written to the program RAM are not appended with any 0s because it is already a full 5 bytes. These zero-extended data fields are appended to a 3-byte field consisting of a 7-bit chip address, a read/write bit, and an 11-bit RAM/register address. The control port knows how many data bytes to expect based on the address that is received in the first three bytes.

The total number of bytes for a single-location write command can vary from four bytes (for a control register write), to eight bytes (for a program RAM write). Burst mode may be used to fill contiguous register or RAM locations. A burst mode write is done by writing the address and data of the first RAM/register location to be written. Rather than ending the control port transaction (by issuing a stop command in I²C mode or by bringing the CLATCH signal high in SPI mode, after the data word), as would be done in a single-address write, the next data word can be written immediately without first writing its specific address. The ADAU1701 control port auto-increments the address of each write, even across the boundaries of the different RAMs and registers. Table 28 and Table 30 show examples of burst mode writes.

Table 27. Parameter RAM Read/Write Format (Single Address)

Byte 0	Byte 1	Byte 2	Byte 3	Bytes 4–6
chip_adr [6:0], $\overline{W}$/R	00000, param_adr[10:8]	param_adr[7:0]	0000, param[27:24]	param [23:0]

Table 28. Parameter RAM Block Read/Write Format (Burst Mode)

Byte 0	Byte 1	Byte 2	Byte 3	Bytes 4–6	Byte 7	Byte 11
chip_adr [6:0], $\overline{W}$/R	00000, param_adr[10:8]	param_adr[7:0]	0000, param[27:24]	param[23:0]	Byte 8	Byte 12
					Byte 9	Byte 13
					Byte 10	Byte 14
			<—param_adr—>		param_adr + 1	param_adr + 2

Table 29. Program RAM Read/Write Format (Single Address)

Byte 0	Byte 1	Byte 2	Bytes 3–7
chip_adr [6:0], $\overline{W}$/R	0000, prog_adr[11:8]	prog_adr[7:0]	prog[39:0]

Table 30. Program RAM Block Read/Write Format (Burst Mode)

Byte 0	Byte 1	Byte 2	Byte 3-7	Byte 8	Byte 13
chip_adr [6:0], $\overline{W}$/R	0000, prog_adr[11:8]	prog_adr[7:0]	prog[39:0]	Byte 9	Byte 14
				Byte 10	Byte 15
				Byte 11	Byte 16
				Byte 12	Byte 17
			<—prog_adr—>	prog_adr +1	prog_adr +2

Table 31. Control Register Read/Write Format (Core, Serial Out 0, Serial Out 1)

Byte 0	Byte1	Byte 2	Byte 3	Byte 4
chip_adr [6:0], $\overline{W}$/R	0000, reg_adr[11:8]	reg_adr[7:0]	data[15:8]	data[7:0]

Table 32. Control Register Read/Write Format (RAM Configuration, Serial Input)

Byte 0	Byte1	Byte 2	Byte 3
chip_adr [6:0], $\overline{W}$/R	0000, reg_adr[11:8]	reg_adr[7:0]	data[7:0]

Table 33. Data Capture Register Write Format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
chip_adr [6:0], $\overline{W}$/R	0000, data_capture_adr[11:8]	data_capture_adr[7:0]	000, progCount[10:6][1]	progCount[5:0][1], regSel[1:0][2]

[1] ProgCount[10:0] = value of program counter where trap occurs (the table of values is generated by the program compiler).

[2] RegSel[1:0] selects one of four registers (see Data Capture Registers section).

Table 34. Data Capture (Control Port Readback) Register Read Format

Byte 0	Byte 1	Byte 2	Bytes 3–5
chip_adr [6:0], $\overline{W}$/R	0000, data_capture_adr[11:8]	data_capture_adr[7:0]	data[23:0]

Table 35. Safeload Address Register Write Format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4
chip_adr [6:0], $\overline{W}$/R	0000, safeload_adr[11:8]	safeload_adr[7:0]	000000, param_adr[9:8]	param_adr[7:0]

Table 36. Safeload Data Register Write Format

Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Bytes 5–7
chip_adr [6:0], $\overline{W}$/R	0000, safeload_adr[11:8]	safeload_adr[7:0]	00000000	0000, data[27:24]	data[23:0]

DATA RAM

The ADAU1701's data RAM is used to store audio data words for processing. For the most part, this process is transparent to the user; the user cannot even directly address this RAM space from the control port. The size of the data RAM is 2 k words.

The user's only concern should be when implementing blocks that utilize large amounts of data RAM space, such as delays. The SigmaDSP core processes delay times in one-sample increments, so the total pool of delay available to the user will equal 2048 × the sample period. For fs = 48 kHz, this means that the pool of available delay is a maximum of about 43 ms. In practice, this much data memory will not be available to the user, though, because every block in a design will use a few data memory locations for its processing. In most DSP programs this will not significantly impact the total delay time. The SigmaStudio compiler will manage the data RAM and will indicate if the number of addresses used has exceeded the maximum available.

MULTIPURPOSE PINS

Table 37. Multipurpose Pin Configuration Registers

Register	Bits[23:20]	Bits[19:16]	Bits[15:12]	Bits[11:8]	Bits[7:4]	Bits[3:0]
MP_CFG0 (2080)	MP5[3:0]	MP4[3:0]	MP3[3:0]	MP2[3:0]	MP1[3:0]	MP0[3:0]
MP_CFG1 (2081)	MP11[3:0]	MP10[3:0]	MP9[3:0]	MP8[3:0]	MP7[3:0]	MP6[3:0]

The ADAU1701 has 12 multipurpose (MP) pins that can be individually programmed to be used as serial data inputs, serial data outputs, digital control inputs and outputs to and from the SigmaDSP core, or as inputs to the four-channel auxiliary ADC. These pins allow the ADAU1701 to be used with external ADCs and DACs, take analog or digital inputs to control settings such as volume control, or output digital signals to drive LED indicators.

MULTIPURPOSE PIN CONFIGURATION REGISTERS

Each multipurpose pin can be set to its different functions from these registers (2080-2081). These two three-byte registers are broken up into twelve 4-bit (nibble) sections that each control a different MP pin as detailed in Table 37. Table 38 lists the different functions of each nibble setting within the MP Pin Configuration Registers. The MSB of each MP pin's 4-bit configuration inverts the input to or output from the pin. The MP pins will have an internal pull-up resistor (approximately 10 kΩ) enabled when they are set to digital inputs (either GPIO input or serial data port input).

Table 38. Multipurpose Pin Configuration Register Bit Functions

MPx[3:0]	Pin Function
1111	Aux ADC input (see Table 40)
1110	Reserved
1101	Reserved
1100	Serial Data Port – inverted (see Table 44)
1011	Open Collector Output - inverted
1010	GPIO Output – inverted
1001	GPIO Input, no debounce – inverted
1000	GPIO Input, debounced – inverted
0111	N/A
0110	Reserved
0101	Reserved
0100	Serial Data Port (see Table 44)
0011	Open Collector Output
0010	GPIO Output
0001	GPIO Input, no debounce
0000	GPIO Input, debounced

GPIO PIN SETTING REGISTER

This register gives the user access through the control port to set the GPIO pins. High or low settings can be directly written to or read from this register after setting bit 7 of the Core Control Register. This register is updated once every LRCLK frame (1/fs)

Table 39. GPIO Pin Setting Register (2056)

Register Bits	Function
15:12	Unused
11	MP11 setting
10	MP10 setting
9	MP9 setting
8	MP8 setting
7	MP7 setting
6	MP6 setting
5	MP5 setting
4	MP4 setting
3	MP3 setting
2	MP2 setting
1	MP1 setting
0	MP0 setting

AUXILIARY ADC

The ADAU1701 has a four-channel auxiliary 8-bit ADC that can be used to connect a potentiometer to control volume, tone, or other parameter settings in the DSP program. Each of the four channels is sampled at the audio sampling frequency (f_S), which defaults to 48 kHz with a 12.288 MHz crystal connected to the ADAU1701 oscillator. Full-scale input on this ADC is 3.3V, so the step size is approximately 13mV (3.3V/256 steps). The input resistance of the ADC is approximate 20 kΩ. Table 40 indicates which four MP pins are mapped to the four channels of the aux ADC. The aux ADC is enabled for those pins by writing 1111 to the appropriate pin's portion of the Multipurpose Pin Configuration Registers.

The auxiliary ADC is turned on by writing a 1 to bit 15 of the Aux ADC enable register (Table 42).

Noise on the ADC input could cause the digital output to be constantly changing by a few LSBs. In cases where the aux ADC is used as a volume control, this would cause small gain fluctuations. To avoid this, a low-pass filter or hysteresis can be added to the aux ADC signal path. These functions can be enabled through the Auxiliary ADC & Power Control Register (2082), shown in Table 41. The filter is enabled by default when the aux ADC is enabled. When data is read from the aux ADC registers, 2 bytes (12 bits of data, plus zero-padded LSBs) are available because of this filtering.

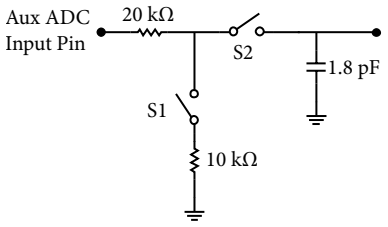


Figure 31. Auxiliary ADC input circuit

Figure 31 shows the input circuit for the auxiliary ADC. Switch S1 enables the aux ADC, and is set by bit 15 of the Aux ADC enable register. The sampling switch, S2, operates at the audio sampling frequency (fs), which is 48 kHz with a 12.288 MHz crystal connected to the chip's oscillator.

The aux ADC registers can be written to directly after bit 8 in the DSP core control register has been set. In this mode, the voltages on the analog inputs are not written into the registers, but rather the data in the registers is written from the control port. These registers take a single byte of data when in control port write mode.

PVDD supplies the 3.3 V power for the aux ADC's analog input. The digital core of the aux ADC is powered with the 1.8 V DVDD signal.

Table 40. Multi-Purpose Pin Aux ADC Mapping

Multipurpose Pin	Function
MP0	N/A
MP1	N/A
MP2	ADC1
MP3	ADC2
MP4	N/A
MP5	N/A
MP6	N/A
MP7	N/A
MP8	ADC3
MP9	ADC0
MP10	N/A
MP11	N/A

Table 41. Auxiliary ADC & Power Control Register (2082)

Register Bits	Function
15:10	Reserved, set to 0
9:8	Aux ADC Filtering 00 = 4-bit hysteresis (12 bit level) 01 = 5-bit hysteresis (12 bit level) 01 = Hysteresis bypassed 11 = Low-pass filter bypassed
7	ADC power-down (both ADCs)
6	Voltage reference buffer power-down
5	Voltage reference power-down
4	Reserved, set to 0
3	DAC0 power-down
2	DAC1 power-down
1	DAC2 power-down
0	DAC3 power-down

Table 42. Aux ADC Enable Register (2084)

Register Bits	Function
15	Enable Auxiliary ADC
14:0	Reserved, set to 0

GENERAL PURPOSE INPUT/OUTPUTS

The general purpose input/output (GPIO) pins can be used as either inputs or outputs. These pins are readable and settable either through the control interface or directly by the SigmaDSP core. When set as inputs, they can be used with push-button switches or rotary encoders to control DSP program settings. Digital outputs may be used to drive LEDs or external logic to indicate the status of internal signals and control other devices. Examples of this use include indicating signal overload, signal present, and button press confirmation.

When set as outputs, these pins can typically drive 5 mA. This is enough current to directly drive high-efficiency LEDs, which typically need 2-4 mA. Standard LEDs require about 20 mA and can be driven from a GPIO output with an external transistor or buffer. When the GPIO pins are set as open-collector outputs, they should be pulled up to a maximum voltage of 3.3 V (the voltage on IOVDD).

SERIAL DATA INPUT/OUTPUT PORTS

The ADAU1701's flexible serial data input and output ports can be set to accept or transmit data in 2-channel formats or in an 8-channel TDM stream. Data is processed in twos complement, MSB-first format. The left channel data field always precedes the right channel data field in the 2-channel streams. In the TDM modes, slots 0 to 3 fall in the first half of the audio frame, and slots 4 to 7 are in the second half of the frame. TDM mode allows fewer multipurpose pins to be used, freeing more pins for other functions. The serial modes are set in the serial output and serial input control registers.

The serial data clocks need to be synchronous with the ADAU1701's master clock input.

The input control register allows control of clock polarity and data input modes. The valid data formats are I^2S , left-justified, right-justified (24-, 20-, 18-, or 16-bit), and 8-channel TDM. In all modes except for the right-justified modes, the serial port will accept an arbitrary number of bits up to a limit of 24. Extra bits will not cause an error, but they will be truncated internally. Proper operation of the right-justified modes requires that there be exactly 64 BCLKs per audio frame. The TDM data is input on SDATA_IN0. The LRCLK in TDM mode can be input to the ADAU1701 either as a 50/50 duty cycle clock or as a bit-wide pulse.

In TDM mode, the ADAU1701 can be a master for 48 kHz and 96 kHz data, but not for 192 kHz data. Table 43 displays the modes in which the serial output port will function.

Table 43. Serial Output Port Master/Slave Mode Capabilities

f_S	2-Channel Modes (I^2S, LJ, RJ)	8-Channel TDM
48 kHz	Master and slave	Master and slave
96 kHz	Master and slave	Master and slave
192 kHz	Master and slave	Slave only

The output control registers give the user control of clock polarities, clock frequencies, clock types, and data format. In all modes except for the right-justified modes (MSB delayed by 8, 12, or 16), the serial port accepts an arbitrary number of bits up to a limit of 24. Extra bits will not cause an error, but will be truncated internally. Proper operation of the right-justified modes requires the LSB to align with the edge of the LRCLK. The default settings of all serial port control registers correspond to 2-channel I^2S mode. All register settings apply to both master and slave modes unless otherwise noted.

The functions of the individual multi-purpose pins in serial data port mode are shown in Table 44. Pins MP0-5 support digital data input to the ADAU1701 and pins MP6-11 handle digital data output from the DSP. The configuration of the serial data input port is set in the Serial Input Control Register (Table 47) and the output port is controlled with the Serial Output Control Register (Table 46). The input port clocks function only as slaves and the output port clocks can be set to be either master or slave. The INPUT_LRCLK and INPUT_BCLK pins (MP4 & MP5) are used to clock the SDATA_INx signals (MP0-3) and the OUTPUT_LRCLK and OUTPUT BCLK (MP10 & MP11) are used to clock the SDATA_OUTx signals (MP6-9).

If an external ADC will be connected as a slave to the ADAU1701, both the input and output port clocks will need to be used. The output LRCLK and BCLK (MP10 & MP11) will need to be set into master mode and connected externally to the input LRCLK and BCLK pins (MP4 & MP5) and the external ADC's clock input pins. The data will be output from the external ADC into the SigmaDSP on one of the four SDATA_IN pins (MP0-3).

Connections to an external DAC are handled exclusively with the output port pins. The output LRCLK and BCLK can be set to be either master or slave, and the SDATA_OUT pins are used to output data from the SigmaDSP to the external DAC.

Table 45 shows the proper configurations for standard audio data formats.

Table 44. Multi-Purpose Pin Serial Data Port Functions

Multipurpose Pin	Function
MP0	SDATA_IN0/TDM_IN
MP1	SDATA_IN1
MP2	SDATA_IN2
MP3	SDATA_IN3
MP4	INPUT_LRCLK (slave only)
MP5	INPUT_BCLK (slave only)
MP6	SDATA_OUT0/TDM_OUT
MP7	SDATA_OUT1
MP8	SDATA_OUT2
MP9	SDATA_OUT3
MP10	OUTPUT_LRCLK (master or slave)
MP11	OUTPUT_BCLK (master or slave)

Table 45. Data Format Configurations

Format	LRCLK Polarity	LRCLK Type	BCLK Polarity	MSB Position
I^2S (Figure 32)	Frame begins on falling edge	Clock	Data changes on falling edge	Delayed from LRCLK edge by one BCLK
Left-Justified (Figure 33)	Frame begins on rising edge	Clock	Data changes on falling edge	Aligned with LRCLK edge
Right-Justified (Figure 34)	Frame begins on rising edge	Clock	Data changes on falling edge	Delayed from LRCLK edge by 8, 12, or 16 BCLKs
TDM with Clock (Figure 35)	Frame begins on falling edge	Clock	Data changes on falling edge	Delayed from start of word clock by one BCLK
TDM with Pulse (Figure 36)	Frame begins on rising edge	Pulse	Data changes on falling edge	Delayed from start of word clock by one BCLK

Table 46. Serial Output Control Register (2078)

Register Bits	Function
15:14	Unused
13	OUTPUT_LRCLK Polarity 0 = Frame Begins on Falling Edge 1 = Frame Begins on Rising Edge
12	OUTPUT_BCLK Polarity 0 = Data Changes on Falling Edge 1 = Data Changes on Rising Edge
11	Master/Slave 0 = Slave 1 = Master
10:9	OUTPUT_BCLK Frequency (Master Mode only) 00 = core_clock/16 01 = core_clock/8 10 = core_clock/4 11 = core_clock/2
8:7	OUTPUT_LRCLK (Master Mode only) 00 = core_clock/1024 01 = core_clock/512 10 = core_clock/256
6	Frame Sync Type 0 = LRCLK 1 = Pulse
5	Serial Output/TDM Mode Control 0 = 8 Serial Data Outputs 1 = Enable TDM on SDATA_OUTx
4:2	MSB Position 000 = Delay by 1 001 = Delay by 0 010 = Delay by 8 011 = Delay by 12 100 = Delay by 16 101 Reserved 111 Reserved
1:0	Output Word Length 00 = 24 Bits 01 = 20 Bits 10 = 16 Bits 11 = Reserved

SERIAL OUTPUT CONTROL REGISTERS

OUTPUT_LRCLK Polarity (Bit 13)

When set to 0, the left channel data is clocked when OUTPUT_LRCLK is low, and the right channel data clocked when OUTPUT_LRCLK is high. When set to 1, the right channel data is clocked when OUTPUT_LRCLK is low, and the left channel data clocked when OUTPUT_LRCLK is high.

OUTPUT_BCLK Polarity (Bit 12)

This bit controls on which edge of the bit clock the output data is clocked. Data changes on the falling edge of OUTPUT_BCLK when this bit is set to 0, and on the rising edge when this bit is set at 1.

Master/Slave (Bit 11)

This bit sets whether the output port is a clock master or slave. The default setting is slave; on power-up, Pins OUTPUT_BCLK and OUTPUT_LRCLK are set as inputs until this bit is set to 1, at which time they become clock outputs.

OUTPUT_BCLK Frequency (Bits 10:9)

When the output port is being used as a clock master, these bits set the frequency of the output bit clock, which is divided down from the internal 49.152 MHz core clock.

OUTPUT_LRCLK Frequency (Bits 8:7)

When the output port is used as a clock master, these bits set the frequency of the output word clock on the OUTPUT_LRCLK pins, which is divided down from the internal 49.152 MHz
core clock.

Frame Sync Type (Bit 6)

This bit sets the type of signal on the OUTPUT_LRCLK pins. When set to 0, the signal is a word clock with a 50% duty cycle; when set to 1, the signal is a pulse with a duration of one bit clock at the beginning of the data frame.

Serial Output/TDM Mode Control (Bit 5)

Setting this bit to 1 changes the output port from multiple serial outputs to a single TDM output stream on the appropriate SDATA_OUTx pin. This bit must be set in both serial output control registers to enable 16-channel TDM on SDATA_OUT0.

MSB Position (Bits 4:2)

These three bits set the position of the MSB of data with respect to the LRCLK edge. The data output of the ADAU1701 is always MSB first.

Output Word Length (Bits 1:0)

These bits set the word length of the output data-word. All bits following the LSB are set to 0.

Table 47. Serial Input Control Register (2079)

Register Bits	Function
7:5	Unused
4	INPUT_LRCLK polarity 0 = Frame begins on falling edge 1 = Frame begins on rising edge
3	INPUT_BCLK polarity 0 = Data changes on falling edge 1 = Data changes on rising edge
2:0	Serial Input Mode 000 = I²S 001 = Left-justified 010 = TDM 011 = Right-justified, 24-bit 100 = Right-justified, 20-bit 101 = Right-justified, 18-bit 110 = Right-justified, 16-bit

SERIAL INPUT CONTROL REGISTER

INPUT_LRCLK Polarity (Bit 4)

When set to 0, the left channel data on the SDATA_INx pins is clocked when INPUT_LRCLK is low; and the right input data clocked when INPUT_LRCLK is high. When set to 1, this is reversed. In TDM mode, when this bit is set to 0, data is clocked in starting with the next appropriate BCLK edge (set in Bit 3 of this register) following a falling edge on the INPUT_LRCLK pin. When set to 1 and running in TDM mode, the input data is valid on the BCLK edge following a rising edge on the word clock (INPUT_LRCLK). INPUT_LRCLK can also operate with a pulse input, rather than a clock. In this case, the first edge of the pulse is used by the ADAU1701 to start the data frame. When this polarity bit is set to 0, a low pulse should be used, and a high pulse should be used when the bit it set to 1.

INPUT_BCLK Polarity (Bit 3)

This bit controls on which edge of the bit clock the input data changes, and on which edge it is clocked. Data changes on the falling edge of INPUT_BCLK when this bit is set to 0, and on the rising edge when this bit is set at 1.

Serial Input Mode (Bits 2:0)

These two bits control the data format that the input port expects to receive. Bits 3 and 4 of this control register will override the settings in Bits 2:0, so all four bits must be changed together for proper operation in some modes. The clock diagrams for these modes are shown in Figure 32, Figure 33, and Figure 34. Note that for left-justified and right-justified modes the LRCLK polarity is high, then low, which is opposite from the default setting of Bit 4.

When these bits are set to accept a TDM input, the ADAU1701's data starts after the edge defined by Bit 4. The ADAU1701's TDM data stream should be input on pin SDATA_IN0. Figure 35 shows a TDM stream with a high-to-low triggered LRCLK and data changing on the falling edge of the BCLK. The ADAU1701 expects the MSB of each data slot delayed by one BCLK from the beginning of the slot, just like in the stereo I²S format. In TDM mode, Channels 0 to 3 will be in the first half of the frame, and Channels 4 to 7 will be in the second half. Figure 36 shows an example of a TDM stream running with a pulse word clock, which would be used to interface to ADI codecs in their auxiliary mode. To work in this mode on either the input or output serial ports, the ADAU1701 should be set to frame beginning on the rising edge of LRCLK, data changing on the falling edge of BCLK, and MSB position delayed from the start of the word clock by one BCLK.

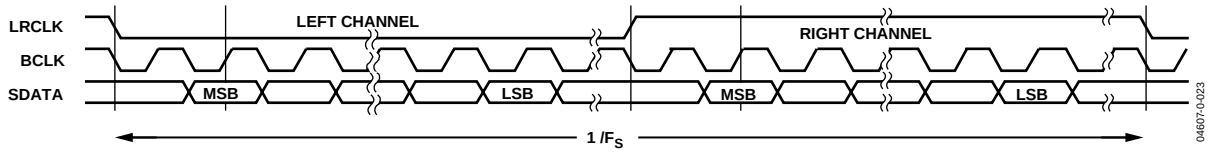


Figure 32. I²S Mode—16 to 24 Bits per Channel

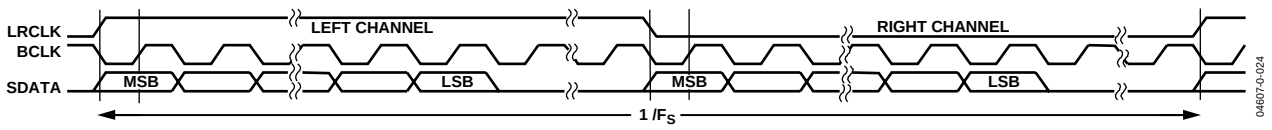


Figure 33. Left-Justified Mode—16 to 24 Bits per Channel

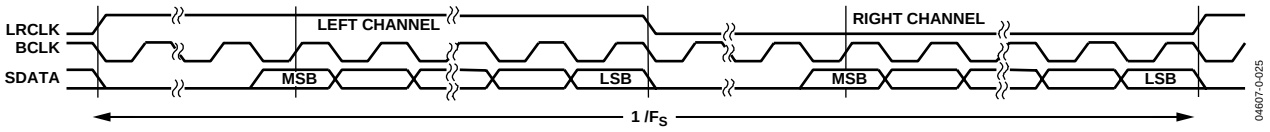


Figure 34. Right-Justified Mode—16 to 24 Bits per Channel

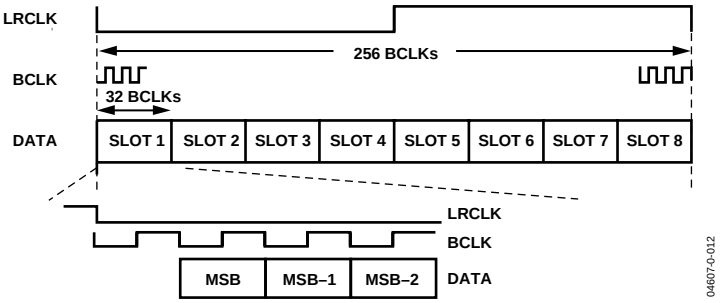


Figure 35. TDM Mode

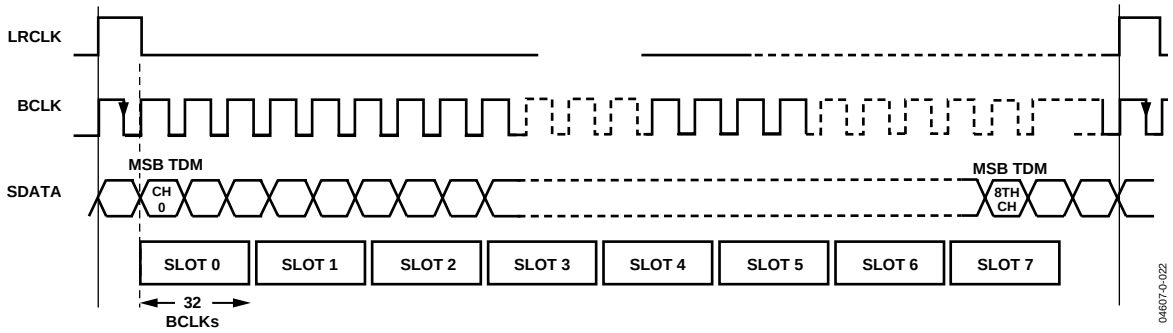


Figure 36. TDM Mode with Pulse Word Clock

LAYOUT RECOMMENDATIONS

PARTS PLACEMENT

The ADC input voltage-to-current resistors and the ADC current set resistor should be placed as close to the input pins (2, 3 & 4) as possible.

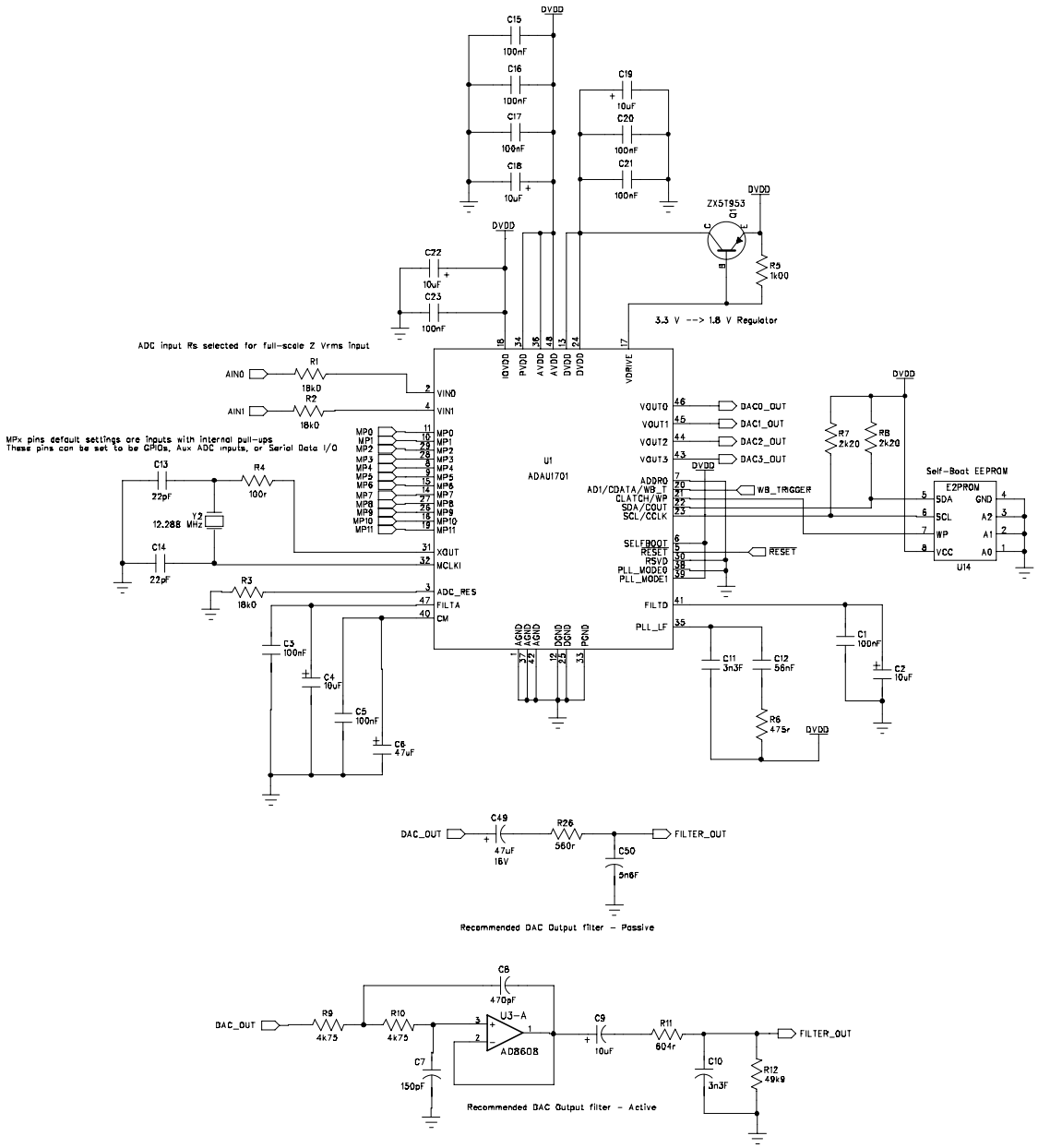
All 100 nF bypass capacitors, which are recommended for every analog, digital, and PLL power/ground pair, should be placed as close to the ADAU1701 as possible. Both the 3.3 V and 1.8 V signals on the board should also each be bypassed with a single bulk capacitor (10-47 µF).

All traces in the crystal oscillator circuit (Figure 16) should be kept as short as possible to minimize stray capacitance. There should not be any long board traces connected to any of these components, as this may affect crystal start-up and operation.

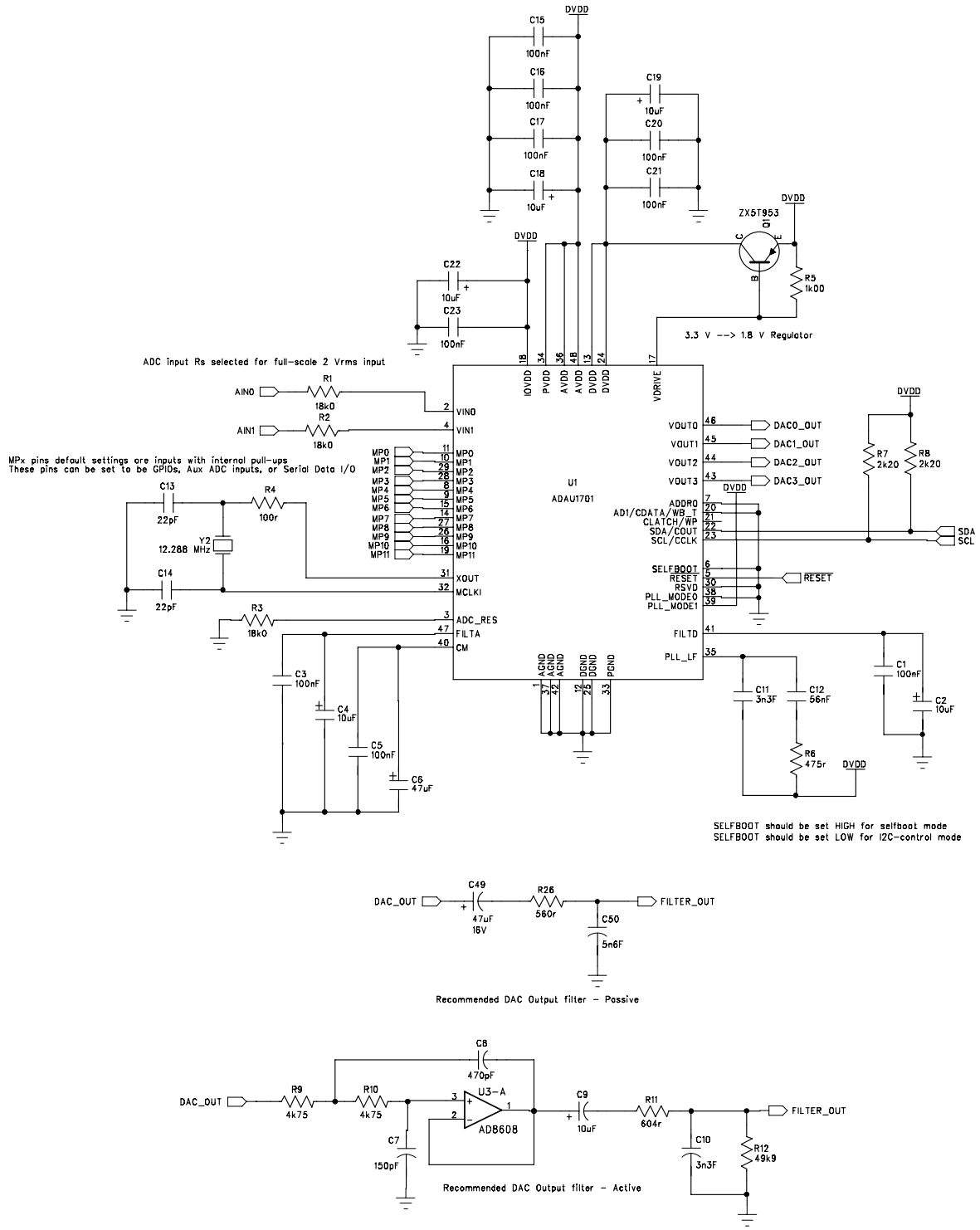
GROUNDING

A single ground plane should be used in the application layout. Components in an analog signal path should still be placed away from digital signals.

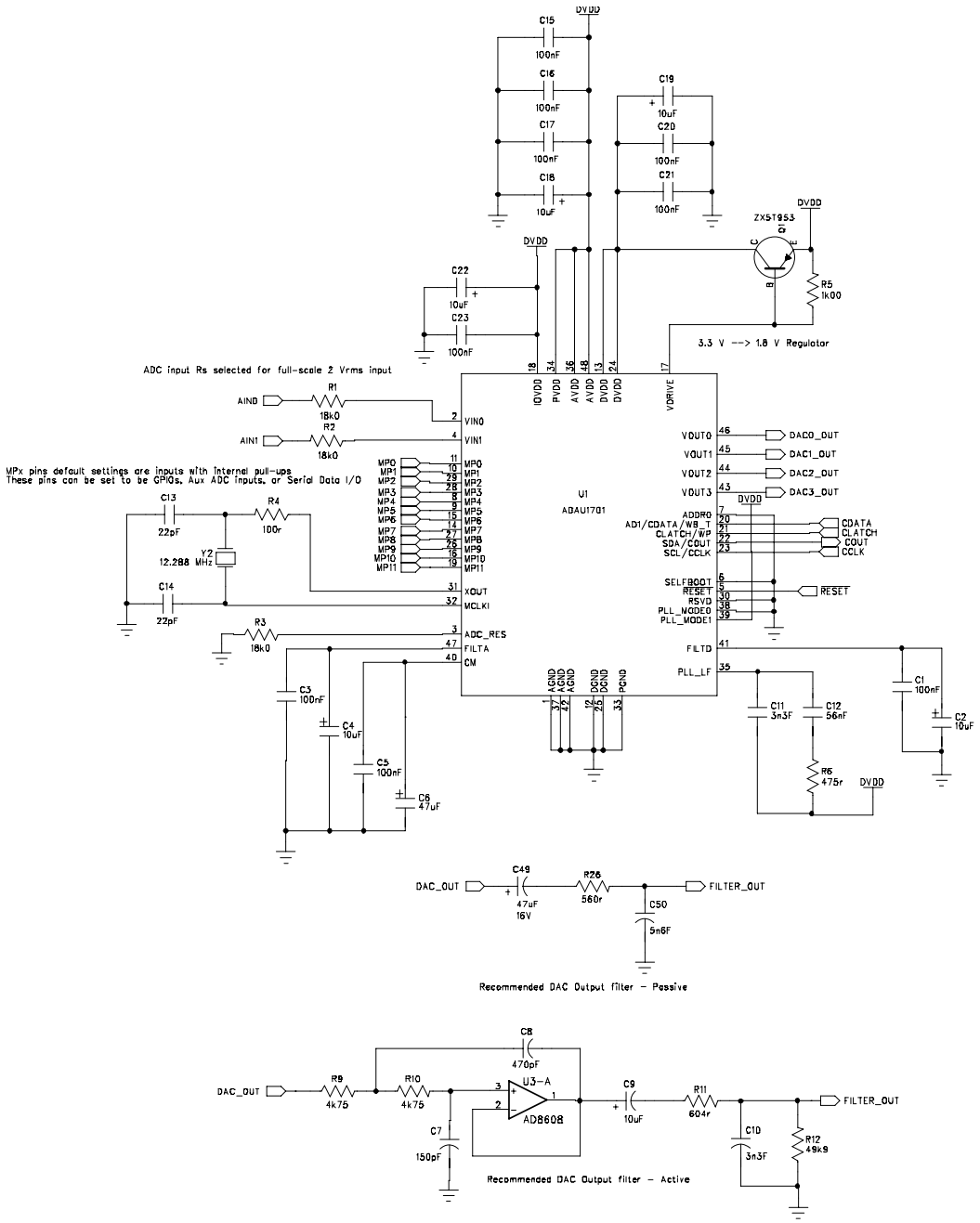
TYPICAL APPLICATION SCHEMATIC – SELF BOOT MODE



TYPICAL APPLICATION SCHEMATIC – I²C CONTROL



TYPICAL APPLICATION SCHEMATIC – SPI CONTROL



DIGITAL TIMING DIAGRAMS

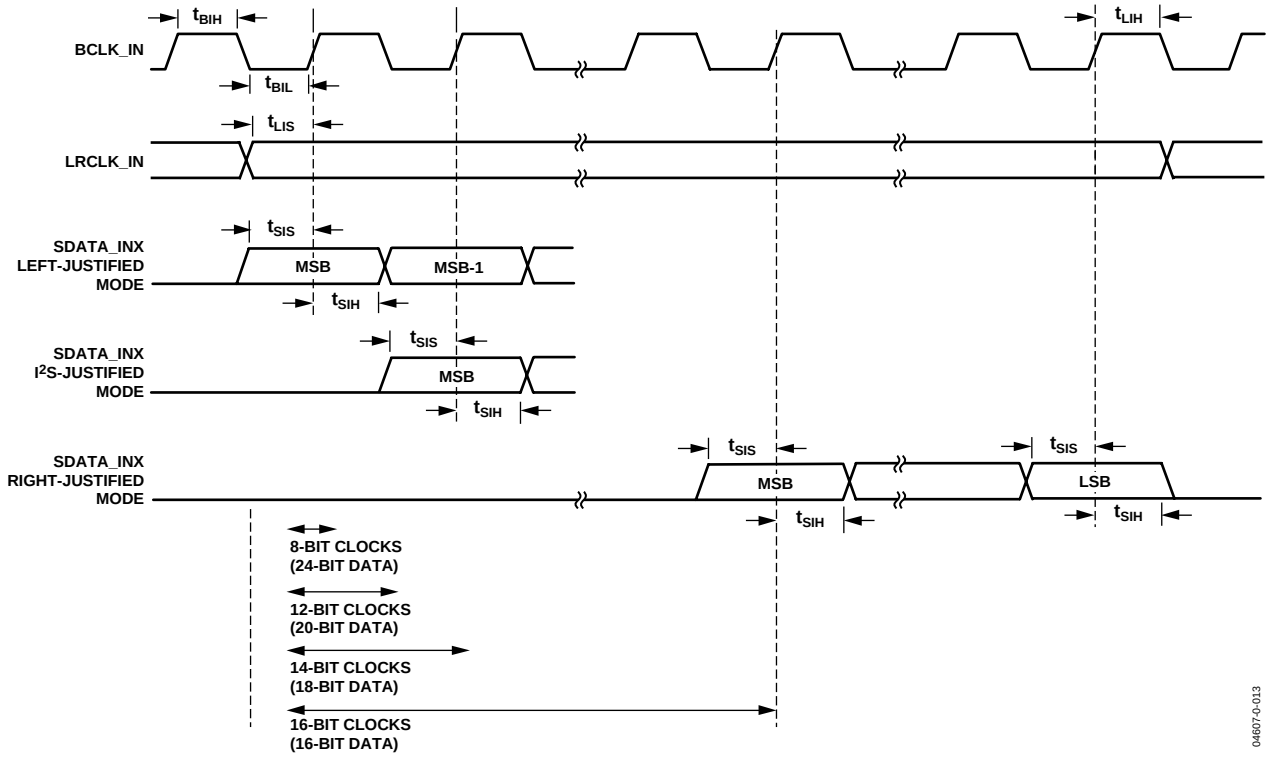


Figure 37. Serial Input Port Timing

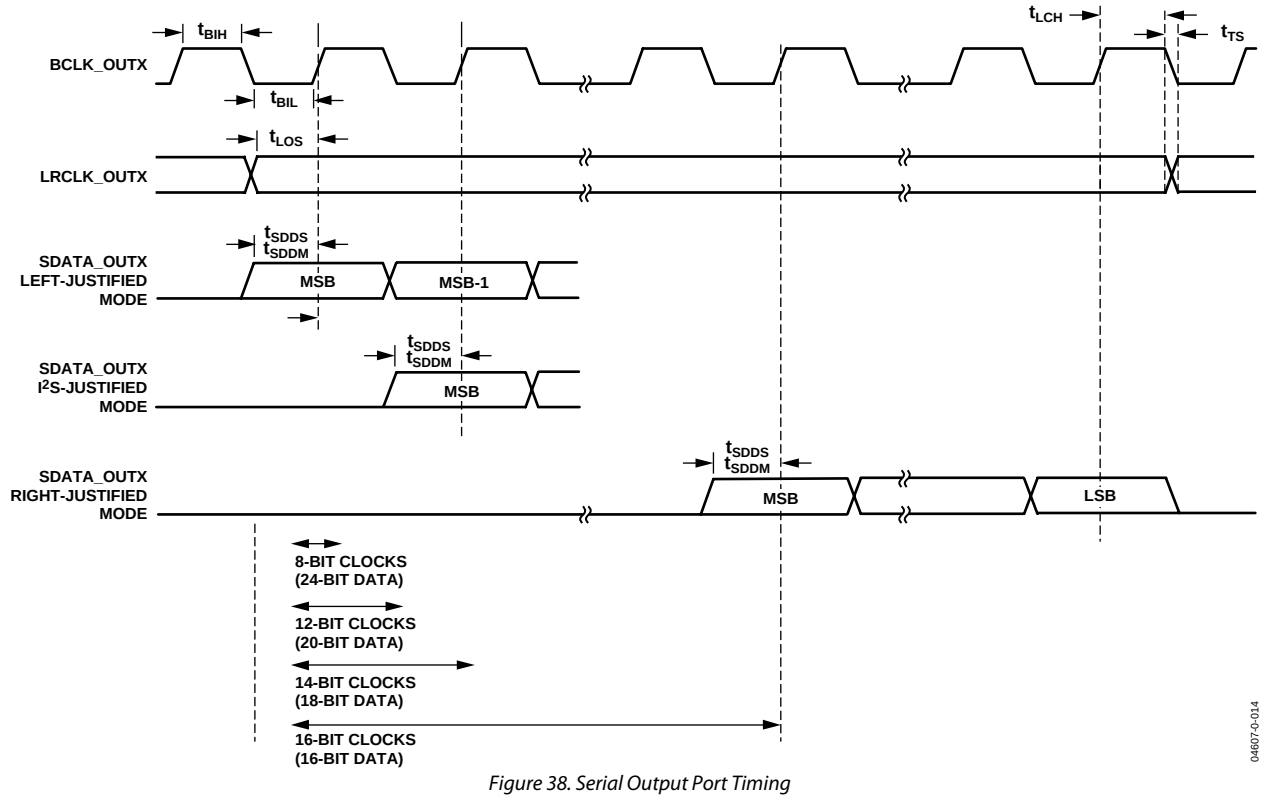


Figure 38. Serial Output Port Timing

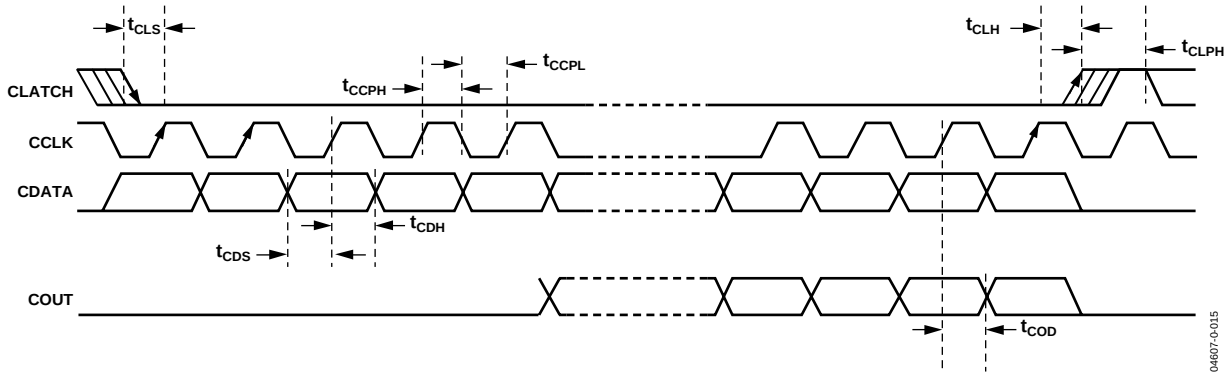


Figure 39. SPI Port Timing

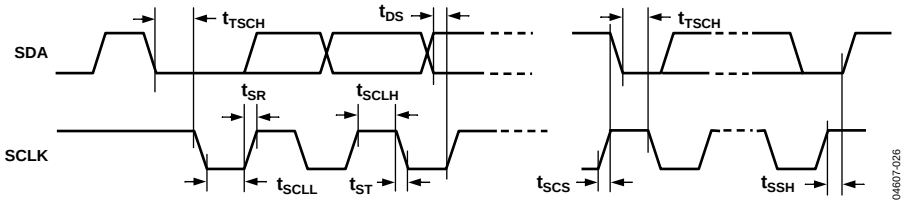


Figure 40. I²C Port Timing

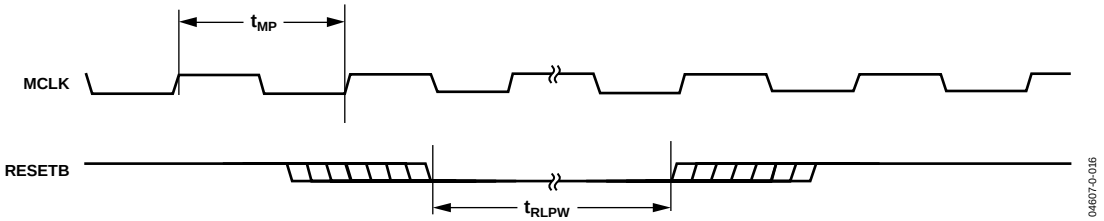


Figure 41. Master Clock and Reset Timing

OUTLINE DIMENSIONS

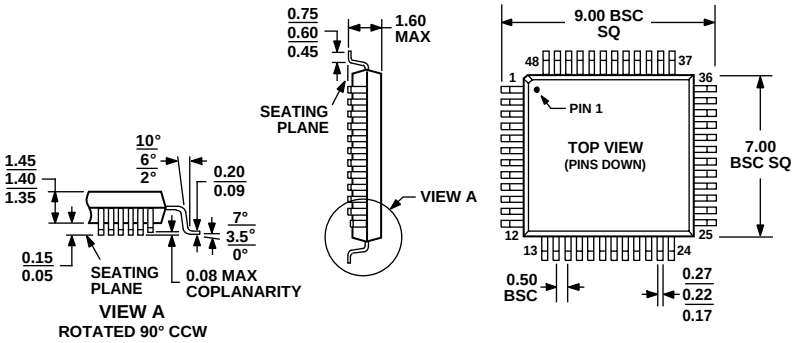


Figure 42. 48-Lead Low-Profile Quad Flat Package [LQFP]
Dimensions Shown in Millimeters

ORDERING GUIDE

Model	Temperature Range	Package Description	Package Option
ADAU1701JSTZ[1]	0°C to 70°C	48-Lead LQFP	ST-48
ADAU1701JSTZ-RL	0°C to 70°C	48-Lead LQFP	ST-48 in 13" Reel
EVAL-ADAU1701EB		Evaluation Board	

[1] Z = Pb-free