Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Active
Type	Sigma
Interface	I²C, SPI
Clock Rate	25MHz
Non-Volatile Memory	-
On-Chip RAM	3kB
Voltage - I/O	3.30V
Voltage - Core	1.80V
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	48-LQFP
Supplier Device Package	48-LQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/analog-devices/adau1702jstz

ADAU1702* PRODUCT PAGE QUICK LINKS

Last Content Update: 02/23/2017

COMPARABLE PARTS

View a parametric search of comparable parts.

EVALUATION KITS

- ADAU1701 Evaluation Board

DOCUMENTATION

Application Notes

- AN-1006: Using the EVAL-ADUSB2EBZ
- AN-923: Designing a System Using the ADAU1701/ ADAU1702 in Self-Boot Mode
- AN-951: Using Hardware Controls with SigmaDSP GPIO Pins

Data Sheet

- ADAU1702: SigmaDSP 28-/56-Bit Audio Processor with Two ADCs and Four DACs Data Sheet

User Guides

- UG-072: Evaluation Board User Guide for ADAU1401

SOFTWARE AND SYSTEMS REQUIREMENTS

- ADAU1701 Sound Audio System Linux Driver
- Firmware Loader for SigmaDSPs

TOOLS AND SIMULATIONS

- SigmaDSP Processors: Software and Tools

DESIGN RESOURCES

- ADAU1702 Material Declaration
- PCN-PDN Information
- Quality And Reliability
- Symbols and Footprints

DISCUSSIONS

View all ADAU1702 EngineerZone Discussions.

SAMPLE AND BUY

Visit the product page to see pricing options.

TECHNICAL SUPPORT

Submit a technical question or find your regional support number.

DOCUMENT FEEDBACK

Submit feedback for this data sheet.

REVISION HISTORY

5/16—Rev. C to Rev. D

Changes to Audio DACs Section and Figure 19 21

6/11—Rev. B to Rev. C

Deleted Table 2 6

2/11—Rev. A to Rev. B

Changes to Test Conditions/Comments Column, Table 3 8
Changes to Table 10 12
Changes to Figure 20, Figure 21 25
Changes to Figure 27 27

9/10—Rev. 0 to Rev. A

Deleted Temperature Range Section and Table 4, Renumbered Sequentially 4
Moved and Changes to Functional Block Diagram Section and Figure 1 4
Changes to Table 1 5
Added Table 2; Renumbered Sequentially 6
Changes to Table 3 and Table 5 7
Changes to Table 7 8
Reorganized Digital Timing Diagrams Section 9
Changes to Figure 2 9
Changes to Figure 5 10
Changes to Table 8 11
Changes to Table 10 12
Changes to Figure 8 to Figure 11 15
Renamed Theory of Operation Section 17
Changes to Initialization Section 18
Changes to Setting the Master Clock/PLL Mode Section 19
Changes to Voltage Regulator Section 20
Changes to Figure 19 22
Changes to Table 14 23
Changes to Figure 22 through Figure 25 26
Deleted Table 20 27
Changes to Self-Boot Section and EEPROM Format Section 28
Added Figure 28, Renumbered Sequentially 29
Changes to Address 2057 to Address 2060 (0x809 to 0x80C)—Auxiliary ADC Data Registers Section 37
Changes to Multipurpose Pins Section and Auxiliary ADC Section 46
Changes to Ordering Guide 56

7/07—Revision 0: Initial Version

DIGITAL INPUT/OUTPUT

Table 2.

Parameter	Symbol	Min	Typ	Max[1]	Unit	Test Conditions/Comments
Input Voltage, High	V_{IH}	2.0		IOVDD	V	
Input Voltage, Low	V_{IL}			0.8	V	
Input Leakage, High	I_{IH}			1	µA	Excluding MCLKI
Input Leakage, Low	I_{IL}			1	µA	Excluding MCLKI and bidirectional pins
Bidirectional Pin Pull-Up Current, Low				150	µA	
MCLKI Input Leakage, High	I_{IH}			3	µA	
MCLKI Input Leakage, Low	I_{IL}			3	µA	
High Level Output Voltage	V_{OH}	2.0			V	I_{OH} = 2 mA
Low Level Output Voltage	V_{OL}			0.8	V	I_{OL} = 2 mA
Input Capacitance				5	pF	
GPIO Output Drive			2		mA	

[1] Maximum specifications are measured across a temperature range of −40°C to +130°C (case), a DVDD range of 1.62 V to 1.98 V, and an AVDD range of 2.97 V to 3.63 V.

POWER

Table 3.

Parameter	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 and PVDD)		50	85	mA
Digital Current (DVDD)		40	60	mA
Analog Current, Reset		35	55	mA
Digital Current, Reset		1.5	4.5	mA
DISSIPATION				
Operation (AVDD, DVDD, PVDD)[2]		286.5		mW
Reset, All Supplies		118		mW
POWER SUPPLY REJECTION RATIO (PSRR)				
1 kHz, 200 mV p-p Signal at AVDD		50		dB

[1] Maximum specifications are measured across a temperature range of −40°C to +130°C (case), a DVDD range of 1.62 V to 1.98 V, and an AVDD range of 2.97 V to 3.63 V.
[2] Power dissipation does not include IOVDD power because the current drawn from this supply is dependent on the loads at the digital output pins.

PLL AND OSCILLATOR

Table 4. PLL and Oscillator

Parameter	Min	Typ	Max[1]	Unit
PLL Operating Range	MCLK_Nom – 20%		MCLK_Nom + 20%	MHz
PLL Lock Time			20	ms
Crystal Oscillator Transconductance (g_m)		78		mmho

[1] Maximum specifications are measured across a temperature range of −40°C to +130°C (case), a DVDD range of 1.62 V to 1.98 V, and an AVDD range of 2.97 V to 3.63 V.

REGULATOR

Table 5. Regulator

Parameter	Min[1]	Typ[1]	Max[1]	Unit
DVDD Voltage	1.7	1.8	1.84	V

[1] Regulator specifications are calculated using a Zetex Semiconductors FZT953 transistor in the circuit.

Digital Timing Diagrams

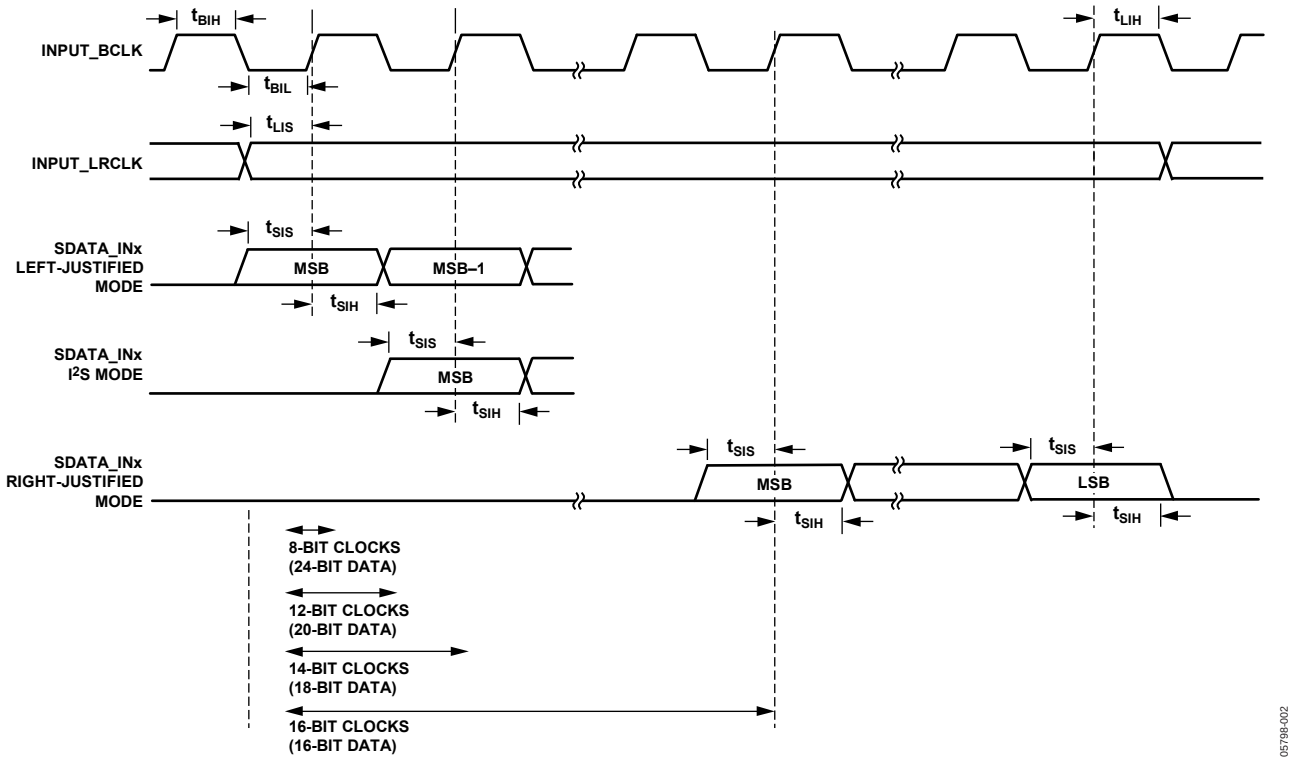

Figure 2. Serial Input Port Timing

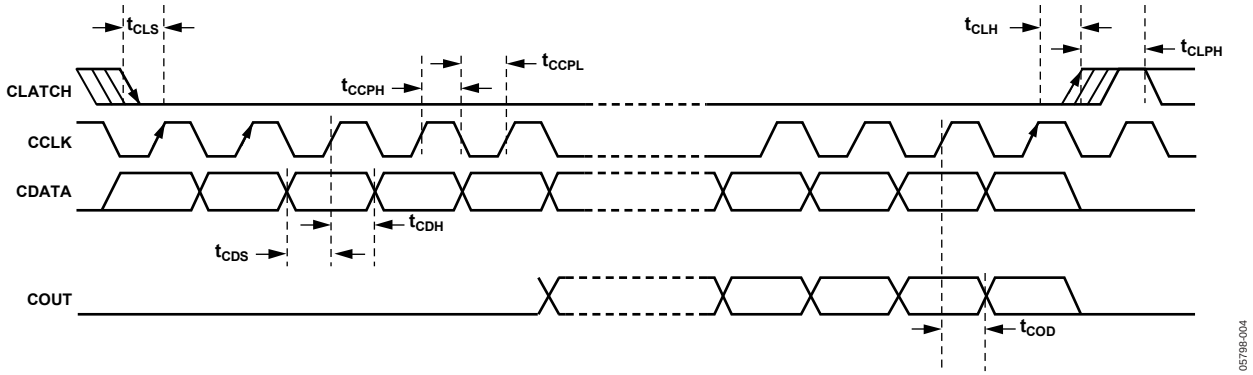

Figure 3. SPI Port Timing

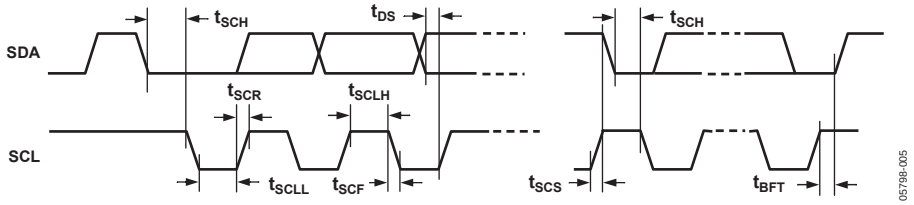

Figure 4. I²C Port Timing

ABSOLUTE MAXIMUM RATINGS

Table 7.

Parameter	Rating
DVDD to Ground	0 V to 2.2 V
AVDD to Ground	0 V to 4.0 V
IOVDD to Ground	0 V to 4.0 V
Digital Inputs	DGND − 0.3 V, IOVDD + 0.3 V
Temperature Range	
Storage	−65°C to +150°C
Operating (Ambient), Functionally Guaranteed	0°C to +70°C
Maximum Junction Temperature	135°C
Soldering (10 sec)	300°C

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

THERMAL RESISTANCE

θ_{JA} is specified for the worst-case conditions, that is, a device soldered in a circuit board for surface-mount packages.

Table 8. Thermal Resistance

Package Type	θ_{JA}	θ_{JC}	Unit
48-Lead LQFP	72	19.5	°C/W

ESD CAUTION

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

Pin No.	Mnemonic	Type[1]	Description
14	MP7	D_IO	Multipurpose GPIO or Serial Output Port Data 1 (SDATA_OUT1). See the Multipurpose Pins section for more details.
15	MP6	D_IO	Multipurpose GPIO, Serial Output Port Data 0, or TDM Data Output (SDATA_OUT0). See the Multipurpose Pins section for more details.
16	MP10	D_IO	Multipurpose GPIO or Serial Output Port LRCLK (OUTPUT_LRCLK). See the Multipurpose Pins section for more details.
17	VDRIVE	A_OUT	Drive for 1.8 V Regulator. The base of the voltage regulator external PNP transistor is driven from VDRIVE. See the Voltage Regulator section for details.
18	IOVDD	PWR	Supply for Input and Output Pins. 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. Always set IOVDD to 3.3 V. The current draw of this pin is variable because it is dependent on the loads of the digital outputs.
19	MP11	D_IO	Multipurpose GPIO or Serial Output Port BCLK (OUTPUT_BCLK). See the Multipurpose Pins section for more details.
20	ADDR1/CDATA/WB	D_IN	I²C Address 1/SPI Data Input/EEPROM Write Back Trigger. This is a multifunction pin as follows: ADDR1: In combination with ADDR0, this sets the I²C address of the IC so that four ADAU1702s can be used on the same I²C bus. See the I²C Port section for details. CDATA: See the SPI Port section for details. WB: A rising (default) or falling (if set in the EEPROM messages) edge on this pin triggers a writeback of the interface registers to the external EEPROM. This function can be used to save parameter data on power-down. See the Self-Boot section for details.
21	CLATCH/WP	D_IO	SPI Latch Signal/ Self-Boot EEPROM Write Protect. This is a multifunction pin as follows: CLATCH: must go low at the beginning of an SPI transaction and high at the end of a transaction. Each SPI transaction can take a different number of cycles on the CCLK pin to complete, depending on the address and read/write bit that are sent at the beginning of the SPI transaction. See the SPI Port section for details. WP: an open-collector output pin when in self-boot mode. The ADAU1702 pulls this low to enable writes to an external EEPROM. This pin should be pulled high to 3.3 V. See the Self-Boot section for details.
22	SDA/COUT	D_IO	I²C Data/ SPI Data Output. This is a multifunction pin, as follows: SDA: this is for I²C data and is a bidirectional open-collector pin. The line connected to this pin should have a 2.2 kΩ pull-up resistor. See the I²C Port section for details. 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. See the SPI Port section for details.
23	SCL/CCLK	D_IO	I²C Clock/SPI Clock. This is a multifunction pin, as follows: SCL: this pin function is for the I²C clock and is always an open-collector input when in I²C control mode. In self-boot mode, this pin is an open-collector output (I²C master). The line connected to this pin should have a 2.2 kΩ pull-up resistor. See the I²C Port section for details. CCLK: this pin function is for the SPI Clock and it either runs continuously or is gated off between SPI transactions. See the SPI Port section for details.
26	MP9	D_IO/A_IO	Multipurpose GPIO, Serial Output Port Data 3 (SDATA_OUT3), or Auxiliary ADC Input 0. See the Multipurpose Pins section for more details.
27	MP8	D_IO/A_IO	Multipurpose GPIO, Serial Output Port Data 2 (SDATA_OUT2), or Auxiliary ADC Input 3. See the Multipurpose Pins section for more details.
28	MP3	D_IO/A_IO	Multipurpose GPIO, Serial Input Port Data 3 (SDATA_IN3), or Auxiliary ADC Input 2. See the Multipurpose Pins section for more details.
29	MP2	D_IO/A_IO	Multipurpose GPIO, Serial Input Port Data 2 (SDATA_IN2), or Auxiliary ADC Input 1. See the Multipurpose Pins section for more details.
30	RSVD	N/A	Reserved. Tie to ground, either directly or through a pull-down resistor.
31	OSCO	D_OUT	Crystal Oscillator Circuit Output. Connect a 100 Ω damping resistor between this pin and the crystal. Do not use this output to directly drive a clock to another IC. If the crystal oscillator is not used, this pin can be left disconnected. See the Using the Oscillator section for details.
32	MCLKI	D_IN	Master Clock Input. MCLKI can either be connected to a 3.3 V clock signal or be the input from the crystal oscillator circuit. See the Setting Master Clock/PLL Mode section for details.

Pin No.	Mnemonic	Type[1]	Description
33	PGND	PWR	PLL Ground Pin. The AGND, DGND, and PGND pins can be tied directly together in a common ground plane. Decouple PGND to PVDD with a 100 nF capacitor.
34	PVDD	PWR	3.3 V Power Supply for the PLL and the Auxiliary ADC Analog Section. Decouple this pin should to PGND with a 100 nF capacitor.
35	PLL_LF	A_OUT	PLL Loop Filter Connection. Two capacitors and a resistor need to be connected to this pin, as shown in Figure 15. See the Setting Master Clock/PLL Mode section for more details.
36, 48	AVDD	PWR	3.3 V Analog Supply. Decouple this pin to AGND with a 100 nF capacitor.
38, 39	PLL_MODE0, PLL_MODE1	D_IN	PLL Mode Setting. 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.
40	CM	A_OUT	1.5 V Common-Mode Reference. Connect a 47 µF decoupling capacitor between this pin and ground to reduce crosstalk between the ADCs and DACs. The material of the capacitors is not critical. This pin can be used to bias external analog circuits, as long as those circuits are not drawing current from the pin (such as when CM is connected to the noninverting input of an op amp).
41	FILTD	A_OUT	DAC Filter Decoupling Pin. Connect a 10 µF capacitor between this pin and ground. The capacitor material is not critical. The voltage on this pin is 1.5 V.
43	VOUT3	A_OUT	VOUT3 DAC Output. The full-scale output voltage is 0.9 V rms. This output can be used with either an active or passive output reconstruction filter. See the Audio DACs section for details.
44	VOUT2	A_OUT	VOUT2 DAC Output. The full-scale output voltage is 0.9 V rms. This output can be used with either an active or passive output reconstruction filter. See the Audio DACs section for details.
45	VOUT1	A_OUT	VOUT1 DAC Output. The full-scale output voltage is 0.9 V rms. This output can be used with either an active or passive output reconstruction filter. See the Audio DACs section for details.
46	VOUT0	A_OUT	VOUT0 DAC Output. The full-scale output voltage is 0.9 V rms. This output can be used with either an active or passive output reconstruction filter. See the Audio DACs section for details.
47	FILTA	A_OUT	ADC Filter Decoupling Pin. Connect a 10 µF capacitor between this pin and ground. The capacitor material is not critical. The voltage on this pin is 1.5 V.

[1] PWR is power/ground, A_IN is analog input, D_IN is digital input, A_OUT is analog output, D_IO is digital input/output, D_IO is digital input/output, A_IO is analog input/output, and N/A is not applicable.

CONTROL PORTS

The ADAU1702 can operate in one of three control modes:

- I²C control
- SPI control
- Self-boot (no external controller)

The ADAU1702 has both a 4-wire SPI control port and a 2-wire I²C bus control port. Each can be used to set the RAMs and registers. When the SELFBOOT pin is low at power-up, the part defaults to I²C mode but can be put into SPI control mode by pulling the CLATCH/WP pin low three times. When the SELFBOOT pin is set high at power-up, the ADAU1702 loads its program, parameters, and register settings from an external EEPROM on startup.

The control port is capable of full read/write operation for all addressable memory 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 registers.

All addresses can be accessed in a single address mode or a burst mode. The first byte (Byte 0) of a control port write contains the 7-bit chip address plus the R/$\overline{W}$ bit. The next two bytes (Byte 1 and Byte 2) together form the subaddress of the memory or register location within the ADAU1702. This subaddress must be two bytes because the memory locations within the ADAU1702 are directly addressable and their sizes exceed the range of single byte addressing. All subsequent bytes (starting with Byte 3) contain the data, such as control port data, program data, or parameter data. The number of bytes per word depends on the type of data that is being written. The exact formats for specific types of writes are shown in Table 20 to Table 29.

The ADAU1702 has several mechanisms for updating signal processing parameters in real time without causing pops or clicks. If large blocks of data need to be downloaded, the output of the DSP core can be halted (using the CR bit in the DSP core control register (Address 2076)), new data can be loaded, and then the device can be restarted. This is typically done during the booting sequence at startup 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 safeload registers. The safeload registers can be used to buffer a full set of parameters (for example, 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 multifunctional, depending on the mode in which the part is operating. Table 13 details these multiple functions.

Table 13. Control Port Pins and SELFBOOT Pin Functions

Pin	I²C Mode	SPI Mode	Self-Boot
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	WB—writeback trigger
CLATCH/WP	Unused input—tie to ground or IOVDD	CLATCH—input	WP—EEPROM write protect, open-collector output
ADDR0	ADDR0—input	ADDR0—input	Unused input—tie to ground or IOVDD

I²C PORT

The ADAU1702 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 ADAU1702 and the system I²C master controller. In I²C mode, the ADAU1702 is always a slave on the bus, meaning it cannot initiate a data transfer. Each slave device is recognized by a unique address. The address byte format is shown in Table 14. The ADAU1702 slave addresses are set with the ADDR0 and ADDR1 pins. The address resides in the first seven bits of the I²C write. The LSB of this 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. Bit 5 and Bit 6 of the address are set by tying the ADDRx pins of the ADAU1702 to Logic Level 0 or Logic Level 1. The full byte addresses, including the pin settings and read/write (R/$\overline{W}$) bit, are shown in Table 15.

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 after a single-word write unless a stop condition is encountered. The registers and RAMs in the ADAU1702 range in width from one to five bytes, so the auto-increment feature knows the mapping between subaddresses and the word length of the destination register (or memory location). A data transfer is always terminated by a stop condition.

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 more than IOVDD (3.3 V).

Table 14. ADAU1702 I²C 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 15. ADAU1702 I²C Addresses

ADDR1	ADDR0	R/$\overline{W}$	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, each device on the I²C bus is in an idle state monitoring the SDA and SCL lines for a start condition and the proper address. The I²C master initiates a data transfer by establishing a start condition, defined by a high-to-low transition on SDA while SCL remains high. This indicates that an address or an address and a data stream follow. All devices on the bus respond to the start condition and shift the next eight bits (the 7-bit address plus the R/$\overline{W}$ bit) MSB first. The device that recognizes the transmitted address responds by pulling the data line low during the ninth clock pulse. This ninth bit is known as an acknowledge bit. All other devices withdraw from the bus at this point and return to the idle condition. The R/$\overline{W}$ bit determines the direction of the data. A Logic 0 on the LSB of the first byte means the master writes information to the peripheral, whereas a Logic 1 means the master reads infor-mation from the peripheral after writing the subaddress and repeating the start address. A data transfer takes place until a stop condition is encountered. A stop condition occurs when SDA transitions from low to high while SCL is held high. Figure 20 shows the timing of an I²C write, and Figure 21 shows an I²C read.

Stop and start conditions can be detected at any stage during the data transfer. If these conditions are asserted out of sequence with normal read and write operations, the ADAU1702 immediately jumps 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 ADAU1702 does not issue an acknowledge and returns to the idle condition. If the user exceeds the highest subaddress while in auto-increment mode, one of two actions is taken. In read mode, the ADAU1702 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. On the other hand, 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 ADAU1702, and the part returns to the idle condition.

SELF-BOOT

On power-up, the ADAU1702 can load a program and a set of parameters that have been 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 ADAU1702 acting as a master on the I²C bus on startup, which occurs when the SELFBOOT pin is set high. The ADAU1702 cannot self-boot in SPI mode.

The maximum necessary EEPROM size for program and parameters is 6688 bytes, or just over 6.5 kB. This does not include register settings or overhead bytes, but such factors do not add a significant number of bytes. This much memory is only needed if the program RAM (512 × five bytes), parameter RAM (1024 × four bytes), and interface registers (8 × four bytes) are completely full. Most applications do not use the full program and parameter RAMs, thus, an 8 kB EEPROM should be sufficient.

A self-boot operation is triggered on the rising edge of $\overline{\text{RESET}}$ when the SELFBOOT and WP pins are set high. The ADAU1702 reads the program, parameters, and register settings from the EEPROM. After the ADAU1702 finishes self-booting, additional messages can be sent to the ADAU1702 on the I²C bus, although this typically is not necessary in a self-booting application. In self-boot mode, the I²C device address is 0x68 for a write and 0x69 for a read. The ADDRx pins have different functions when the chip is in this mode, so the settings on them can be ignored.

The ADAU1702 does not self-boot if WP is set low. Holding this pin low allows the EEPROM to be programmed in-circuit. The WP pin is pulled low (it typically has a resistor pull-up) to enable writes to the EEPROM, but this in turn disables the self-boot function until the WP pin is returned high.

The ADAU1702 is a master on the I²C bus during self-boot and writeback. Although it is uncommon for an application using self-boot to also have a microcontroller connected to the control lines, care should be taken that no other device tries to write to the I²C bus during self-boot or writeback. The ADAU1702 generates SCL at 8 × f_S; therefore, for a f_S of 48 kHz, SCL runs at 384 kHz. SCL has a duty cycle of 3/8 in accordance with the I²C specification.

The ADAU1702 reads from EEPROM Chip Address 0xA1. The LSBs of the addresses of some EEPROMs are pin configurable; in most cases, these pins should be tied low to set this address.

EEPROM Format

The EEPROM data contains a sequence of messages. Each discrete message is one of the seven types defined in Table 18 and consists of a sequence of one or more bytes. The first byte identifies the message type. Bytes are written MSB first. Most messages are block write (0x01) types, which are used for writing to the ADAU1702 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. As with all other control port transactions, following the chip address is a two-byte register/memory address field.

Figure 28 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 operation (no op) bytes (Line 2 to Line 4) so that the interface register data appears on Page 2 of the EEPROM. Next, follows the write header (Line 4) and then 32 bytes of interface register data (Line 5 to Line 8). Finally, the program RAM data, starting at Address 0x04, 0x00 is written (Line 9 to Line 11). In this example, the program length is 70 words, or 350 bytes; therefore, 332 additional bytes are included in the EEPROM but are not shown in Figure 28.

Writeback

A writeback occurs when the WB pin is triggered and data is written to the EEPROM from the ADAU1702. This function is typically used to save the volume setting and other parameter settings to the EEPROM immediately prior to power being removed from the system. A rising edge on the WB pin triggers a writeback when the device is in self-boot mode, unless a message to set the WB to the falling edge sensitive (0x05) is contained in the self-boot message sequence. Only one writeback takes place unless a message to set multiple writebacks (0x04) is contained in the self-boot message sequence. The WP pin is pulled low when a writeback is triggered to allow writing to the EEPROM.

The ADAU1702 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 core control register. The parameter settings that the user wants to save are configured in SigmaStudio.

SIGNAL PROCESSING

The ADAU1702 is designed to provide all audio signal processing functions commonly used in stereo or multichannel playback systems. The signal processing flow is designed using the 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 data. The input and output word lengths of the DSP core are 24 bits. Four extra headroom bits are used in the processor to allow internal gains of up to 24 dB without clipping. Additional gains can be achieved by initially scaling down the input signal in the DSP signal flow.

NUMERIC FORMATS

DSP systems commonly use a standard numeric format. 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 ADAU1702 uses the same numeric format for both the parameter and data values. The format follows in the Numerical Format: 5.23 section.

Numerical Format: 5.23

Linear 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 0011 0011 0011 0011 0011 = −0.1
1111 1111 1111 1111 1111 1111 1111 = (1 LSB below 0.0)
0000 0000 0000 0000 0000 0000 0000 = 0.0
0000 0000 1100 1100 1100 1100 1101 = 0.1
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 DSP core. This allows internal gains of up to 24 dB without internal clipping.

A digital clipper circuit is used between the output of the DSP core and the DACs or serial port outputs (see Figure 29). 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 29 shows the maximum signal levels at each point in the data flow in both binary and decibel levels.

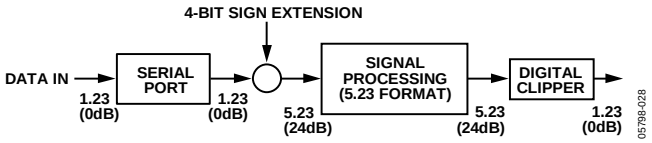

Figure 29. Numeric Precision and Clipping Structure

PROGRAMMING

On power-up, the ADAU1702 default program passes the unprocessed input signals to the outputs (shown in Figure 13), but the outputs are muted by default (see the Power-Up Sequence section). There are 512 instruction cycles per audio sample, resulting in about 25 MIPS available. The SigmaDSP runs in a stream-oriented manner, meaning that all 512 instructions are executed each sample period. The ADAU1702 can also be set up to accept double- or quad-speed inputs by reducing the number of instructions per sample that are set in the core control register.

The part can be easily programmed using SigmaStudio (Figure 30), a graphical tool provided by Analog Devices. No knowledge of writing line-level DSP code is required. More information about SigmaStudio can be found at www.analog.com.

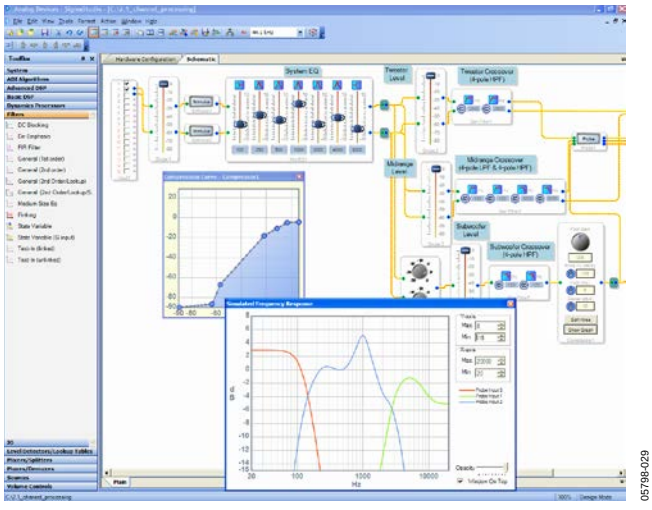

Figure 30. SigmaStudio Screen Shot

Register Address		No. of		MSB															LSB	
				D31	D30	D29	D28	D27	D26	D25	D24	D39	D38	D37	D36	D35	D34	D33	D32	
Hex	Dec	Bytes	Name	D15	D14	D13	D12	D11	D10	D9	D8	D23 D7	D22 D6	D21 D5	D20 D4	D19 D3	D18 D2	D17 D1	D16 D0	Default
0x0820	2080	3	MP Pin Config. 0[23:16]									MP53	MP52	MP51	MP50	MP43	MP42	MP41	MP40	0x00
			MP Pin Config. 0[15:0]	MP33	MP32	MP31	MP30	MP23	MP22	MP21	MP20	MP13	MP12	MP11	MP10	MP03	MP02	MP01	MP00	0x0000
0x0821	2081	3	MP Pin Config. 1[23:16]									MP113	MP112	MP111	MP110	MP103	MP102	MP101	MP100	0x00
			MP Pin Config. 1[15:0]	MP93	MP92	MP91	MP90	MP83	MP82	MP81	MP80	MP73	MP72	MP71	MP70	MP63	MP62	MP61	MP60	0x0000
0x0822	2082	2	Auxiliary ADC and power control	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	FIL1	FIL0	AAPD	VBPD	VRPD	RSVD	D0PD	D1PD	D2PD	D3PD	0x0000
0x0823	2083	2	Reserved	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	0x0000
0x0824	2084	2	Auxiliary ADC enable	AAEN	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	0x0000
0x0825	2085	2	Reserved	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	0x0000
0x0826	2086	2	Oscillator power-down	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	OPD	RSVD	RSVD	0x0000
0x0827	2087	2	DAC setup	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	DS1	DS0	0x0000

[1] Shading indicates that registers do not fill these locations; therefore, control bits do not exist in these locations.

CONTROL REGISTER DETAILS

ADDRESS 2048 TO ADDRESS 2055 (0x0800 TO 0x0807)—INTERFACE REGISTERS

The interface registers are used in self-boot mode to save parameters that need to be written to the external EEPROM. The ADAU1702 then recalls these parameters from the EEPROM after the next reset or power-up. Therefore, system parameters such as volume and EQ settings can be saved during power-down and recalled the next time the system is turned on.

There are eight 32-bit interface registers, which allow eight 28-bit (plus zero padding) parameters to be saved. The parameters to be saved in these registers are selected 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 ADAU1702 to write the current contents of the interface registers to the EEPROM. See the Self-Boot section for details.

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

Table 31. Interface Register Bit Map

D31 D15	D30 D14	D29 D13	D28 D12	D27 D11	D26 D10	D25 D9	D24 D8	D23 D7	D22 D6	D21 D5	D20 D4	D19 D3	D18 D2	D17 D1	D16 D0	Default
0	0	0	0	IF27	IF26	IF25	IF24	IF23	IF22	IF21	IF20	IF19	IF18	IF17	IF16	0x0000
IF15	IF14	IF13	IF12	IF11	IF10	IF09	IF08	IF07	IF06	IF05	IF04	IF03	IF02	IF01	IF00	0x0000

Table 32.

Bit Name	Description
IF[27:0]	Interface register 28-bit parameter

ADDRESS 2057 TO ADDRESS 2060 (0x0809 TO 0x080C)—AUXILIARY ADC DATA REGISTERS

These registers hold the data generated by the 4-channel auxiliary ADC. The ADCs have eight bits of precision and can be extended to 12 bits if filtering is selected via Bits FIL[1:0] of the auxiliary ADC and power control register. The SigmaDSP program reads this data as a 1.11 format data-word with a range of 0 to 1.0. This data-word is mapped to the 5.23 format parameter word with the four MSBs and 12 LSBs set to 0. A full-scale code of 255 results in a value of 1.0 in 5.23 format. These registers can be written to directly if the auxiliary ADC data registers control port write (AACW) mode bit is set in the DSP core control register.

Table 35. Auxiliary ADC Data Register Bit Map

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	Default
0	0	0	0	AA11	AA10	AA09	AA08	AA07	AA06	AA05	AA04	AA03	AA02	AA01	AA00	0x0000

Table 36.

Bit Name	Description
AA[11:0]	Auxiliary ADC output data, MSB first

ADDRESS 2074 TO ADDRESS 2075 (0x081A TO 0x081B)—DATA CAPTURE REGISTERS

The ADAU1702 data capture feature allows the data at any node in the signal processing flow to be sent to one of two readable registers. This feature is useful for monitoring and displaying 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 is to occur. The register select field programs one of four registers in the DSP core that transfers this information to the data capture register when the program counter reaches this step.

The captured data is in 5.19, twos complement data format, which comes from the internal 5.23 data-word with the four LSBs truncated.

The data that must be written to set up the data capture is a concatenation of the 10-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 Location 2074 and Location 2075. The format for writing and reading to the data capture registers is shown in Table 26 and Table 27.

Table 42. Safeload Data Registers Bit Map

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	Default
0	0	0	0	PC09	PC08	PC07	PC06	PC05	PC04	PC03	PC02	PC01	PC00	RS01	RS00	0x0000

Table 43.

Bit Name	Description	
PC[9:0]	10-bit program counter address	
RS[1:0]	Select the register to be transferred to the data capture output	
	Setting	**Function**
	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)

ADDRESS 2076 (0x081C)—DSP CORE CONTROL REGISTER

Table 44. DSP Core Control Register Bit Map

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	Default
RSVD	RSVD	GD1	GD0	RSVD	RSVD	RSVD	AACW	GPCW	IFCW	IST	ADM	DAM	CR	SR1	SR0	0x0000

Table 45. DSP Core Control Register

Bit Name	Description	
GD[1:0]	GPIO debounce control. These bits set the debounce time of the multipurpose pins, which are set as GPIO inputs.	
	Setting	**Time (ms)**
	00	20
	01	40
	10	10
	11	5
AACW	Auxiliary ADC data registers control port write mode. Setting this bit allows data to be written directly to the auxiliary ADC data registers (Address 2057 to Address 2060) from the control port. When this bit is set, the auxiliary ADC data registers ignore the settings on the multipurpose pins.	
GPCW	GPIO pin setting register control port write mode. When this bit is set, the GPIO pin setting register (Address 2056) can be written to directly from the control port and this register ignores the input settings on the multi-purpose pins.	
IFCW	Interface registers control port write mode. When this bit is set, data can be written directly to the interface registers (Address 2048 to Address 2055) from the control port. In that state, the interface registers are not written from the SigmaDSP program.	
IST	Initiate safeload transfer. Setting this bit to 1 initiates a safeload transfer to the parameter RAM. This bit is automatically cleared when the operation is complete. 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.	
ADM	Mute ADCs. This bit mutes the output of the ADCs. The bit defaults to 0 and is active low; therefore, it must be set to 1 to transmit audio signals from the ADCs.	
DAM	Mute DACs. This bit mutes the output of the DACs. The bit defaults to 0 and is active low; therefore, it must be set to 1 to transmit audio signals from the DACs.	
CR	Clear internal registers to 0. This bit defaults to 0 and is active low. It must be set to 1 for a signal to pass through the SigmaDSP core.	
SR[1:0]	Sample rate. These bits set the number of DSP instructions for every sample and the sample rate at which the ADAU1702 operates. At the default setting of 1×, there are 512 instructions per audio sample. This setting should be used with sample rates such as 48 kHz and 44.1 kHz. At the 2× setting, the number of instructions per frame is halved to 256 and the ADCs and DACs nominally run at a 96 kHz sample rate. At the 4× setting, there are 128 instructions per cycle and the converters run at a 192 kHz sample rate.	
	Setting	**Function**
	00	1× (512 instructions)
	01	2× (256 instructions)
	10	4× (128 instructions)
	11	Reserved

ADDRESS 2079 (0x081F)—SERIAL INPUT CONTROL REGISTER

Table 48. Serial Input Control Register Bit Map

D7	D6	D5	D4	D3	D2	D1	D0	Default
0	0	0	ILP	IBP	M2	M1	M0	0x00

Table 49. Serial Input Control Register Bit Descriptions

Bit Name	Description	
ILP	INPUT_LRCLK polarity. When this bit is set to 0, the left channel data on the SDATA_INx pins is clocked when INPUT_LRCLK is low and the right channel data is clocked when INPUT_LRCLK is high. When this bit is set to 1, the clocking of these channels 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) after a falling edge on the INPUT_LRCLK pin. When this bit is set to 1 and the device is running in TDM mode, the input data is valid on the BCLK edge after 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 ADAU1702 to start the data frame. When this polarity bit is set to 0, use a low pulse; when the bit it set to 1, use a high pulse.	
IBP	INPUT_BCLK polarity. 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.	
M[2:0]	Serial input mode. These two bits control the data format that the input port expects to receive. Bit 3 and Bit 4 of this control register override the settings of Bits[2:0]; therefore, 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 and then low, which is the opposite of the default setting for ILP. When these bits are set to accept a TDM input, the ADAU1702 data starts after the edge defined by ILP. Input the ADAU1702 TDM data stream 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 ADAU1702 expects the MSB of each data slot to be delayed by one BCLK from the beginning of the slot, as it would in stereo I²S format. In TDM mode, Channel 0 to Channel 3 are in the first half of the frame, and Channel 4 to Channel 7 are in the second half. Figure 36 shows an example of a TDM stream running with a pulse word clock, which is used to interface to ADI codecs in auxiliary mode. To work in this mode with either the input or output serial ports, set the ADAU1702 to begin the frame on the rising edge of LRCLK, to change data on the falling edge of BCLK, and to delay the MSB position from the start of the word clock by one BCLK.	
	Setting	**Function**
	000	I²S
	001	Left justified
	010	TDM
	011	Right justified, 24 bits
	100	Right justified, 20 bits
	101	Right justified, 18 bits
	110	Right justified, 16 bits
	111	Reserved

MULTIPURPOSE PINS

The ADAU1702 has 12 multipurpose (MP) pins that can be individually programmed for use as serial data inputs, serial data outputs, digital control inputs/outputs to and from the SigmaDSP core, or inputs to the 4-channel auxiliary ADC. These pins allow the ADAU1702 to be used with external ADCs and DACs. They also use analog or digital inputs to control settings such as volume control, or use output digital signals to drive LED indicators. Every MP pin has an internal 15 kΩ pull-up resistor.

AUXILIARY ADC

The ADAU1702 has a 4-channel, auxiliary, 8-bit ADC that can be used in conjunction with 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). Full-scale input on this ADC is 3.0 V, thus the step size is approximately 12 mV (3.0 V/256 steps). The input resistance of the ADC is approximately 30 kΩ. Table 61 indicates which four MP pins are mapped to the four channels of the auxiliary ADC. The auxiliary ADC is enabled for those pins by writing 1111 to the appropriate portion of the multipurpose pin configuration registers.

The auxiliary ADC is turned on by setting the AAEN bit of the auxiliary ADC enable register (see Table 56).

Noise on the ADC input can cause the digital output to change constantly by a few LSBs. If the auxiliary ADC is used to control volume, this constant change causes small gain fluctuations. To avoid this, add a low-pass filter or hysteresis to the auxiliary ADC signal path by enabling either function in the auxiliary ADC and power control register (Address 2082), as described in Table 54. The filter is enabled by default when the auxiliary ADC is enabled. When data is read from the auxiliary ADC registers, two bytes (12 bits of data plus zero padded LSBs) are available because of this filtering.

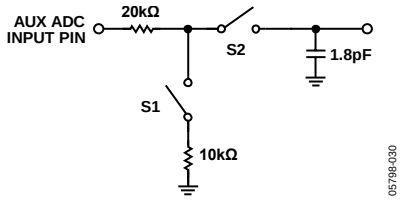

Figure 31. Auxiliary ADC Input Circuit

Figure 31 shows the input circuit for the auxiliary ADC. Switch S1 enables the auxiliary ADC and is set by Bit 15 of the auxiliary ADC enable register. The sampling switch, S2, operates at the audio sampling frequency.

The auxiliary ADC data registers can be written to directly after AACW 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.

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

Table 61. Multipurpose Pin Auxiliary ADC Mapping

Multipurpose Pin	Function
MP0	Not applicable
MP1	Not applicable
MP2	ADC1
MP3	ADC2
MP4	Not applicable
MP5	Not applicable
MP6	Not applicable
MP7	Not applicable
MP8	ADC3
MP9	ADC0
MP10	Not applicable
MP11	Not applicable

GENERAL-PURPOSE INPUT/OUTPUT PINS

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

When set as an output, each pin can typically drive 2 mA. This is enough current to directly drive some high efficiency LEDs. Standard LEDs require about 20 mA of current and can be driven from a GPIO output with an external transistor or buffer. Because of issues that can arise from simultaneously driving or sinking a large current on many pins, take care in the application design to avoid connecting high efficiency LEDs directly to many or all of the MPx pins. If many LEDs are required, use an external driver.

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 flexible serial data input and output ports of the ADAU1702 can be set to accept or transmit data in 2-channel format 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 TDM mode, Slot 0 to Slot 3 are in the first half of the audio frame, and Slot 4 to Slot 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.

LAYOUT RECOMMENDATIONS

PARTS PLACEMENT

Place the ADC input voltage-to-current resistors and the ADC current set resistor as close as possible to the two, three, and four input pins.

All 100 nF bypass capacitors, which are recommended for every analog, digital, and PLL power/ground pair, should be placed as close as possible to the ADAU1702. Bypass each of the 3.3 V and 1.8 V signals on the board with a single bulk capacitor (10 µF to 47 µF).

To minimize stray capacitance, keep all traces in the crystal oscillator circuit (Figure 14) as short as possible. In addition, avoid long board traces connected to any of these components because such traces can affect crystal startup and operation.

GROUNDING

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

SPI CONTROL

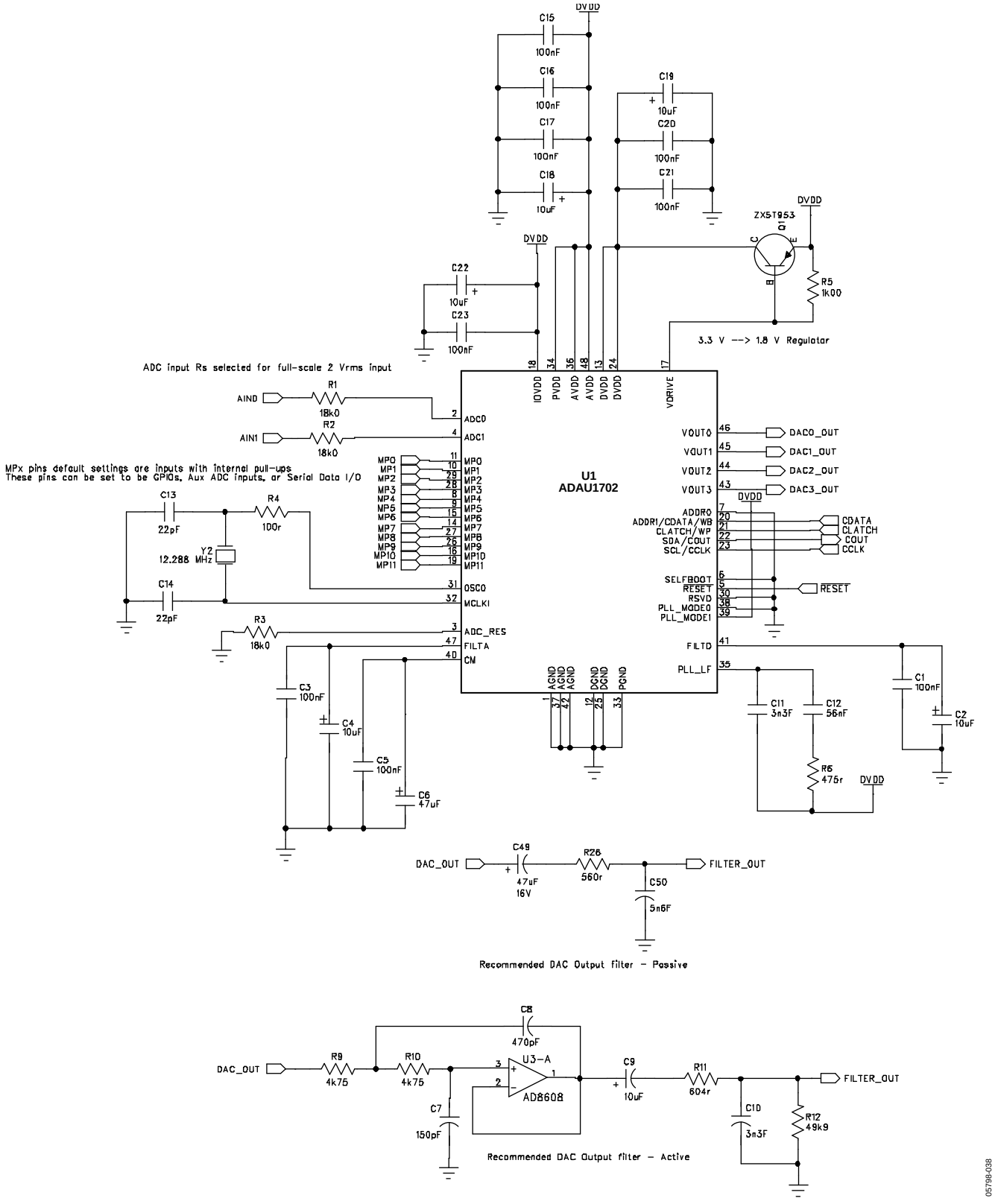

Figure 39. SPI Control Schematic