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	50MHz
Non-Volatile Memory	-
On-Chip RAM	12kB
Voltage - I/O	3.30V
Voltage - Core	1.80V
Operating Temperature	-40°C ~ 105°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/adau1401awbstz

POWER

Table 4.

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)		25	40	mA
Analog Current, Reset		35	55	mA
Digital Current, Reset		1.5	4.5	mA
DISSIPATION				
Operation (AVDD, DVDD, PVDD)[2]		259.5		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.

TEMPERATURE RANGE

Table 5.

Parameter	Min	Typ	Max	Unit
Functionality Guaranteed	−40		+105	°C ambient

PLL AND OSCILLATOR

Table 6.

Parameter[1]	Min	Typ	Max	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 7.

Parameter[1]	Min	Typ	Max	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 SPECIFICATIONS

Table 8.

Parameter[1]	Limit t_{MIN}	Limit t_{MAX}	Unit	Description
MASTER CLOCK				
t_{MP}	36	244	ns	MCLKI period, 512 × f_S mode.
t_{MP}	48	366	ns	MCLKI period, 384 × f_S mode.
t_{MP}	73	488	ns	MCLKI period, 256 × f_S mode.
t_{MP}	291	1953	ns	MCLKI period, 64 × f_S mode.
SERIAL PORT				
t_{BIL}	40		ns	INPUT_BCLK low pulse width.
t_{BIH}	40		ns	INPUT_BCLK high pulse width.
t_{LIS}	10		ns	INPUT_LRCLK setup; time to INPUT_BCLK rising.
t_{LIH}	10		ns	INPUT_LRCLK hold; time from INPUT_BCLK rising.
t_{SIS}	10		ns	SDATA_INx setup; time to INPUT_BCLK rising.
t_{SIH}	10		ns	SDATA_INx hold; time from INPUT_BCLK rising.
t_{LOS}	10		ns	OUTPUT_LRCLK setup in slave mode.
t_{LOH}	10		ns	OUTPUT_LRCLK hold in slave mode.
t_{TS}		5	ns	OUTPUT_BCLK falling to OUTPUT_LRCLK timing skew.
t_{SODS}		40	ns	SDATA_OUTx delay in slave mode; time from OUTPUT_BCLK falling.
t_{SODM}		40	ns	SDATA_OUTx delay in master mode; time from OUTPUT_BCLK falling.
SPI PORT				
f_{CCLK}		6.25	MHz	CCLK frequency.
t_{CCPL}	80		ns	CCLK pulse width low.
t_{CCPH}	80		ns	CCLK pulse width high.
t_{CLS}	0		ns	CLATCH setup; time to CCLK rising.
t_{CLH}	100		ns	CLATCH hold; time from CCLK rising.
t_{CLPH}	80		ns	CLATCH pulse width high.
t_{CDS}	0		ns	CDATA setup; time to CCLK rising.
t_{CDH}	80		ns	CDATA hold; time from CCLK rising.
t_{COD}		101	ns	COUT delay; time from CCLK falling.
I²C PORT				
f_{SCL}		400	kHz	SCL frequency.
t_{SCLH}	0.6		µs	SCL high.
t_{SCLL}	1.3		µs	SCL low.
t_{SCS}	0.6		µs	SCL setup time, relevant for repeated start condition.
t_{SCH}	0.6		µs	SCL hold time. After this period, the first clock is generated.
t_{DS}	100		ns	Data setup time.
t_{SCR}		300	ns	SCL rise time.
t_{SCF}		300	ns	SCL fall time.
t_{SDR}		300	ns	SDA rise time.
t_{SDF}		300	ns	SDA fall time.
t_{BFT}	0.6			Bus-free time; time between stop and start.
MULTIPURPOSE PINS AND RESET				
t_{GRT}		50	ns	GPIO rise time.
t_{GFT}		50	ns	GPIO fall time.
t_{GIL}		1.5 × 1/f_S	µs	GPIO input latency; time until high/low value is read by core.
t_{RLPW}	20		ns	$\overline{\text{RESET}}$ low pulse width.

[1] All timing specifications are given for the default (I²S) states of the serial input port and the serial output port (see Table 66).

PIN CONFIGURATION AND FUNCTION DESCRIPTIONS

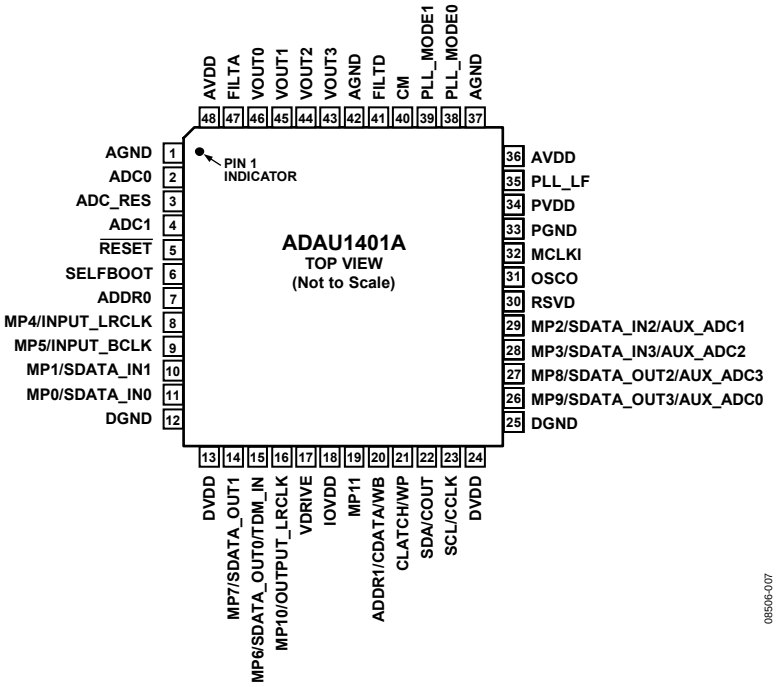

Figure 7. 48-Lead LQFP Pin Configuration

Table 11. Pin Function Descriptions

Pin No.	Mnemonic	Type[1]	Description
1, 37, 42	AGND	PWR	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	ADC0	A_IN	Analog Audio Input 0. Full-scale 100 µA rms input. The current input allows the input voltage level to be scaled with an external resistor. An 18 kΩ resistor results in a 2 V rms full-scale input. See the Audio ADCS section for details.
3	ADC_RES	A_IN	ADC Reference Current. The full-scale current of the ADCs can be set with an external 18 kΩ resistor connected between this pin and ground. See the Audio ADCS section for details.
4	ADC1	A_IN	Analog Audio Input 1. Full-scale 100 µA rms input. The current input allows the input voltage level to be scaled with an external resistor. An 18 kΩ resistor results in a 2 V rms full-scale input.
5	$\overline{\text{RESET}}$	D_IN	Active Low Reset Input. Reset is triggered on a high-to-low edge, and the ADAU1401A exits reset on a low-to-high edge. For more information about initialization, see the Power-Up Sequence section for details.
6	SELFBOOT	D_IN	Enable/Disable Self-Boot. SELFBOOT selects control port (low) or self-boot (high). Setting this pin high initiates a self-boot operation when the ADAU1401A is brought out of a reset. This pin can be tied directly to the control voltage or pulled up/down with a resistor. See the Self-Boot section.
7	ADDR0	D_IN	I^2C and SPI Address 0. In combination with ADDR1, this pin allows up to four ADAU1401A devices to be used on the same I^2C bus or up to two ICs to be used with a common SPI CLATCH signal. See the I^2C Port section for details.
8	MP4/INPUT_LRCLK	D_IO	Multipurpose GPIO/Serial Input Port LRCLK. See the Multipurpose Pins section for more details.
9	MP5/INPUT_BCLK	D_IO	Multipurpose GPIO/Serial Input Port BCLK. See the Multipurpose Pins section for more details.
10	MP1/SDATA_IN1	D_IO	Multipurpose GPIO/Serial Input Port Data 1. See the Multipurpose Pins section for more details.
11	MP0/SDATA_IN0	D_IO	Multipurpose GPIO/Serial Input Port Data 0. See the Multipurpose Pins section for more details.
12, 25	DGND	PWR	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, 24	DVDD	PWR	1.8 V Digital Supply. The input for this pin can be supplied either externally or generated

THEORY OF OPERATION

The core of the ADAU1401A is a 28-bit DSP (56-bit with double-precision processing) optimized for audio processing. The program and parameter RAMs can be loaded with a custom audio processing signal flow built using the SigmaStudio graphical programming software from Analog Devices, Inc. The values stored in the parameter RAM control individual signal processing blocks, such as equalization filters, dynamics processors, audio delays, and mixer levels. A safeload feature allows transparent parameter updates and prevents clicks in the output signals.

The program RAM, parameter RAM, and register contents can be saved in an external EEPROM, from which the ADAU1401A can self-boot on startup. In this standalone mode, parameters can be controlled through the on-board multipurpose pins. The ADAU1401A 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 again on power-up.

The ADAU1401A can operate with digital or analog inputs and outputs, or a mix of both. The stereo ADC and four DACs each have an SNR of at least +100 dB and a THD + N of at least −83 dB. The 8-channel, flexible serial data input/output ports allow glueless interconnection to a variety of ADCs, DACs, general-purpose DSPs, S/PDIF receivers and transmitters, and sample rate converters. The serial ports of the ADAU1401A can be configured in I²S, left-justified, right-justified, or TDM serial port compatible modes.

Twelve multipurpose pins (MP0 to MP11) allow the ADAU1401A to receive external control signals as input and to output flags or controls to other devices in the system. The MPx pins can be configured as digital I/Os, inputs to the 4-channel auxiliary ADC, or serial data I/O ports. As inputs, these pins can be connected to buttons, switches, rotary encoders, potentiometers, IR receivers, or other external circuitry to control the internal signal processing program. When configured as outputs, these pins can be used to drive LEDs, control other ICs, or connect to other external circuitry in an application.

The ADAU1401A 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 configuration and serial modes of the chip. The ADAU1401A can be configured for either SPI or I²C control, or it can self-boot from an external EEPROM.

An on-board oscillator can be connected to an external crystal to generate the master clock. In addition, a master clock phase-locked loop (PLL) allows the ADAU1401A to be clocked from various 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 internal master clock of the core.

The SigmaStudio software is used to program and control the SigmaDSP® through the control port. Along with designing and tuning a signal flow, SigmaStudio tools can be used to configure all of the DSP registers and burn a new program into the external EEPROM. The SigmaStudio 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. In addition, the interface provides enough flexibility and programmability for an experienced DSP programmer to have in-depth control of the design. In SigmaStudio, the user can connect graphical blocks (such as biquad filters, dynamics processors, mixers, and delays), compile the design, and load the program and parameter files into the ADAU1401A memory through the control port. Signal processing blocks available in the provided libraries include

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

Additional processing blocks are always being developed. Analog Devices also provides proprietary and third-party algorithms for applications such as matrix decoding, bass enhancement, and surround virtualizers. Contact Analog Devices for information about licensing these algorithms.

The ADAU1401A 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. The ADAU1401A is fabricated on a single monolithic, integrated circuit and is packaged in a 48-lead LQFP for operation over the −40°C to +105°C temperature range.

AUDIO ADCs

The ADAU1401A has two Σ-Δ ADCs. The signal-to-noise ratio (SNR) of the ADCs is 100 dB, and the THD + N is −83 dB.

The stereo audio ADCs are current input; therefore, 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 be scaled to provide the proper full-scale current input. The ADC0 and ADC1 input pins, as well as the ADC_RES pin, have an internal 2 kΩ resistor for ESD protection. The voltage seen directly on the ADC input pins is the 1.5 V common-mode voltage.

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 μA rms with an external 18 kΩ resistor on ADC_RES (20 kΩ total, because it is in series with the internal 2 kΩ). The only reason to change the ADC_RES resistor is if a sampling rate other than 48 kHz is used.

The voltage-to-current resistors connected to ADC0 and ADC1 set the full-scale voltage input of the ADCs. With a full-scale current input of 100 μA rms, a 2.0 V rms signal with an external 18 kΩ resistor (in series with the 2 kΩ internal resistor) results in an input using the full range of the ADC. The matching of these resistors to the ADC_RES resistor is important to the operation of the ADCs. For these three resistors, a 1% tolerance is recommended.

The ADC0 input pin and/or the ADC1 input pin can be left unconnected if the corresponding channel of the ADC is unused.

The calculations of resistor values assume a 48 kHz sample rate. The recommended input and current setting resistors scale linearly 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 as follows:

$$R_{TOTAL} = 20\ \text{k}\Omega \times \frac{48{,}000}{f_{S\_NEW}}$$

The values of the resistors (internal plus external) in series with the ADC0 and ADC1 pins can be calculated as follows:

$$R_{INPUT\ TOTAL} = (rms\ Input\ Voltage) \times 10\ \text{k}\Omega \times \frac{48{,}000}{f_{S\_NEW}}$$

Table 14 lists the external and total resistor values for common signal input levels at a 48 kHz sampling rate. A full-scale rms input voltage of 0.9 V is shown in the table because a full-scale signal at this input level is equal to a full-scale output on the DACs.

Table 14. ADC Input Resistor Values

Full-Scale RMS Input Voltage (V)	ADC_RES Value (kΩ)	ADC0/ADC1 Resistor Value (kΩ)	Total ADC0/ADC1 Input Resistance (External + Internal) (kΩ)
0.9	18	7	9
1.0	18	8	10
2.0	18	18	20

Figure 17 shows a typical configuration of the ADC inputs for a 2.0 V rms input signal for a f_S of 48 kHz. The 47 μF capacitors are used to ac-couple the signals so that the inputs are biased at 1.5 V.

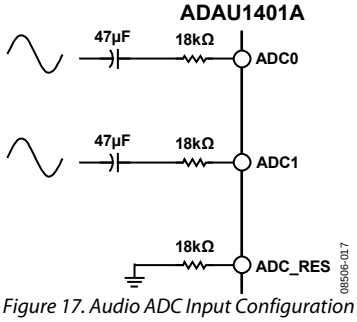

Figure 17. Audio ADC Input Configuration

AUDIO DACs

The ADAU1401A includes four Σ-Δ DACs. The SNR of the DACs is 104 dB, and the THD + N is −90 dB. A full-scale output on the DACs is 0.9 V rms (2.5 V p-p).

The DACs are in an inverting configuration. If a signal inversion from input to output is undesirable, it can be reversed either by using an inverting configuration for 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 passive reconstruction filter. A single-pole, passive, low-pass filter with a 50 kHz corner frequency, as shown in Figure 18, is sufficient to filter the DAC out-of-band noise, although an active filter may provide better audio performance. Figure 19 shows a triple-pole, active, low-pass filter that provides a steeper roll-off and better stop-band attenuation than the passive filter. In this configuration, the V+ and V− pins of the AD8606 op amp are set to VDD and ground, respectively.

To properly initialize the DACs, the DS[1:0] bits in the DAC setup register (Address 2087) should be set to 01.

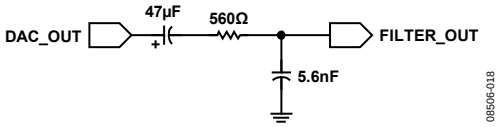

Figure 18. Passive DAC Output Filter

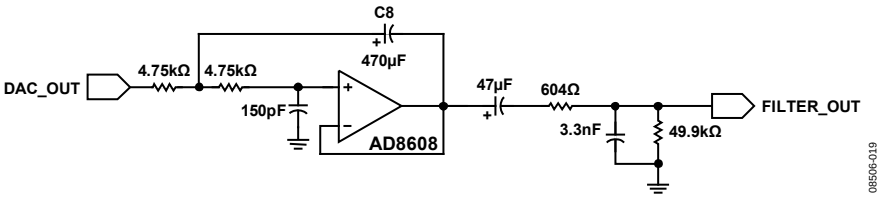

Figure 19. Active DAC Output Filter

I²C PORT

The ADAU1401A 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 ADAU1401A and the system I²C master controller. In I²C mode, the ADAU1401A 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 16. The ADAU1401A 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 ADAU1401A 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 17.

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 ADAU1401A range in width from one to five bytes; therefore, 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 16. ADAU1401A 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 17. ADAU1401A 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 that the master writes information to the peripheral, whereas a Logic 1 means that the master reads information from the peripheral after writing the subaddress and repeating the start address. A data transfer takes place until a stop condition is encountered. A stop condition occurs when SDA transitions from low to high while SCL is held high. Figure 20 shows the timing of an I²C write, and Figure 21 shows the timing of 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 ADAU1401A 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 ADAU1401A 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 ADAU1401A 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 ADAU1401A, and the part returns to the idle condition.

I²C Read and Write Operations

Figure 22 shows the timing of a single-word write operation. On every ninth clock, the ADAU1401A issues an acknowledge by pulling SDA low.

Figure 23 shows the timing of a burst mode write sequence. This figure shows an example where the target destination registers are two bytes. The ADAU1401A knows to increment its subaddress register every two bytes because the requested subaddress corresponds to a register or memory area with a word length of two bytes.

The timing of a single-word read operation is shown in Figure 24. Note that the first R/$\overline{W}$ bit is 0, indicating a write operation. This is because the subaddress still needs to be written to set up the internal address. After the ADAU1401A acknowledges the receipt of the subaddress, the master must issue a repeated start command followed by the chip address byte with the R/$\overline{W}$ bit set to 1 (read). This causes the ADAU1401A SDA to reverse and begin driving data back to the master. The master then responds every ninth pulse with an acknowledge pulse to the ADAU1401A.

Figure 25 shows the timing of a burst mode read sequence. This figure shows an example where the target read registers are two bytes. The ADAU1401A increments its subaddress register every two bytes because the requested subaddress corresponds to a register or memory area with word lengths of two bytes. Other addresses may have word lengths ranging from one to five bytes. The ADAU1401A always decodes the subaddress and sets the auto-increment circuit so that the address increments after the appropriate number of bytes.

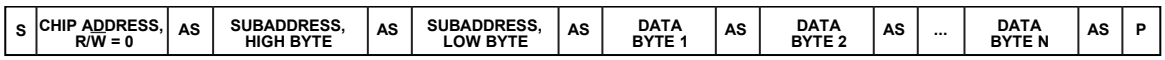

Figure 22. Single-Word I²C Write Format

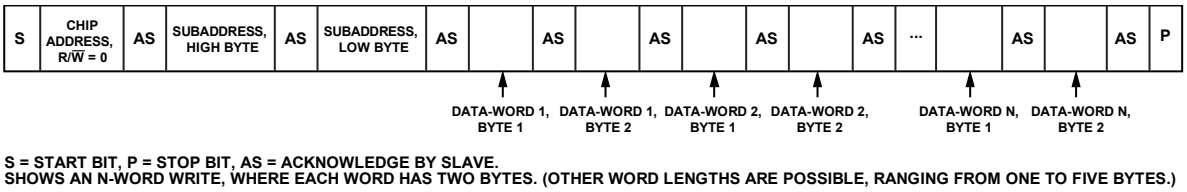

Figure 23. Burst Mode I²C Write Format

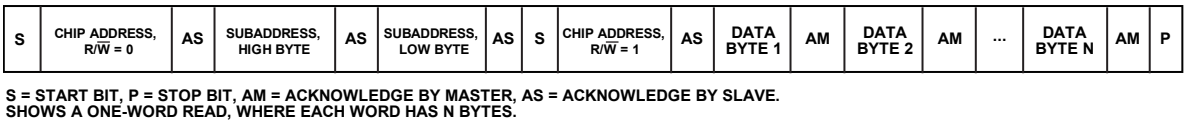

Figure 24. Single-Word I²C Read Format

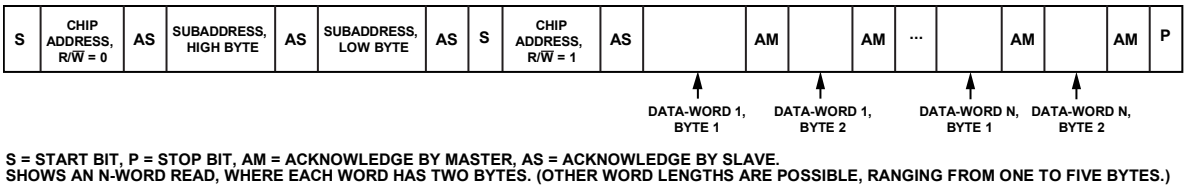

Figure 25. Burst Mode I²C Read Format

SELF-BOOT

On power-up, the ADAU1401A 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-boot is accomplished by the ADAU1401A acting as a master on the I²C bus on startup, which occurs when the SELFBOOT pin is set high. The ADAU1401A cannot self-boot in SPI mode.

The maximum necessary EEPROM size for program and parameters is 9248 bytes, or just over 8.5 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 needed only if the program RAM (1024 × 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 ADAU1401A reads the program, parameters, and register settings from the EEPROM. After the ADAU1401A finishes self-booting, additional messages can be sent to the ADAU1401A on the I²C bus, although this typically is not necessary in a self-booting 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 can be ignored.

The ADAU1401A 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 ADAU1401A 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 ADAU1401A generates SCL at 8 × f_S; therefore, for a f_S of 48 kHz, SCL is 384 kHz. SCL has a duty cycle of 3/8 in accordance with the I²C specification.

The ADAU1401A 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 20 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 ADAU1401A 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 2-byte register/memory address field.

Figure 28 shows an example of what should be stored in the EEPROM, starting with EEPROM Address 0x00. In this example, the interface registers are first set to control port write mode (see Line 1 of Figure 28), which is followed by 18 no-operation (no-op) bytes (see Line 2 to Line 4 of Figure 28) so that the interface register data appears on Page 2 of the EEPROM. Next follows the write header, which comprises a write, length and device address (see Line 4 of Figure 28), and then 32 bytes of interface register data (see Line 5 to Line 8 of Figure 28). Finally, the program RAM data, starting at ADAU1401A Address 0x04 0x00 is written (see Line 9 to Line 11 of Figure 28). In this example, the program length is 70 words, or 350 bytes, so 332 more 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 ADAU1401A. This function is typically used to save the volume setting 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 device is in self-boot mode, unless a message to set the WB pin to be 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 ADAU1401A can only write 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 DSP core control register. The parameter settings that should be saved are configured in SigmaStudio.

SIGNAL PROCESSING

The ADAU1401A 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 ADAU1401A uses the same numeric format for both the parameter and data values. The format is as described 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 indicates the maximum signal levels at each point in the data flow in both binary and decibel levels.

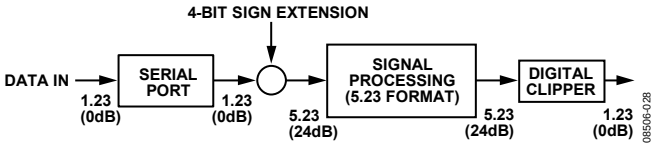

Figure 29. Numeric Precision and Clipping Structure

PROGRAMMING

On power-up, the ADAU1401A 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 1024 instruction cycles per audio sample, resulting in about 50 MIPS being available. The SigmaDSP runs in a stream-oriented manner, meaning that all 1024 instructions are executed each sample period. The ADAU1401A can also be set to accept double- or quad-speed inputs by reducing the number of instructions per sample that are set in the DSP core control register.

The part can be easily programmed using SigmaStudio (see 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.

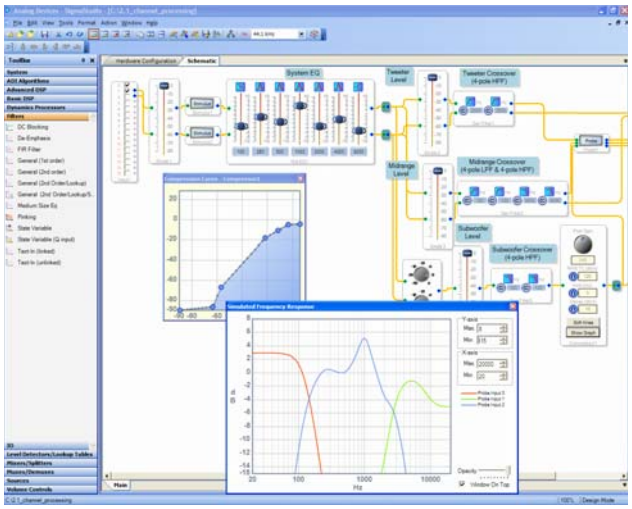

Figure 30. SigmaStudio Screen Shot

Register Address		No. of Bytes	Name	MSB															LSB	Default
				D31	D30	D29	D28	D27	D26	D25	D24	D39	D38	D37	D36	D35	D34	D33	D32	
												D23	D22	D21	D20	D19	D18	D17	D16	
Hex	Dec			D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
0x081C	2076	2	DSP core control	RSVD	RSVD	GD1	GD0	RSVD	RSVD	RSVD	AACW	GPCW	IFCW	IST	ADM	DAM	CR	SR1	SR0	0x0000
0x081D	2077	1	Reserved									RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	RSVD	0x00
0x081E	2078	2	Serial output control	0	0	OLRP	OBP	M/S	OBF1	OBF0	OLF1	OLF0	FST	TDM	MSB2	MSB1	MSB0	OWL1	OWL0	0x0000
0x081F	2079	1	Serial input control									0	0	0	ILP	IBP	M2	M1	M0	0x00
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

ADDRESS 2056 (0x0808)—GPIO PIN SETTING REGISTER

This register allows the user to set the GPIO pins through the control port. High or low settings can be directly written to or read from this register after setting the GPIO pin setting register control port write mode bit (GPCW) in the DSP core control register. This register is updated once every LRCLK frame ($1/f_S$).

Table 35. GPIO Pin Setting Register Bit Map

D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	Default
0	0	0	0	MP11	MP10	MP09	MP08	MP07	MP06	MP05	MP04	MP03	MP02	MP01	MP00	0x0000

Table 36. GPIO Pin Setting Register Bit Descriptions

Bit Name	Description
MP[11:0]	Setting of the corresponding multipurpose pin when controlled through SPI or I^2C

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

The ADAU1401A 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 an integer 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, which 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 28 and Table 29.

Table 44. Data Capture 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 45. Data Capture Registers Bit Descriptions

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	
	Settings	**Function**
	00	Select the Multiplier X input (MULT_X_INPUT) register
	01	Select the Multiplier Y input (MULT_Y_INPUT) register
	10	Select the multiplier-accumulator output (MAC_OUT) register
	11	Select the accumulator feedback (ACCUM_FBACK) register

ADDRESS 2079 (0x081F)—SERIAL INPUT CONTROL REGISTER

Table 50. 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 51. 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 ADAU1401A to start the data frame. When this polarity bit is set to 0, a low pulse should be used; when the bit it set to 1, a high pulse should be used.	
IBP	INPUT_BCLK polarity. This bit controls on which bit clock edge 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 to 1.	
M[2:0]	Serial input mode. These three 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 five 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 the ILP bit. When these bits are set to accept a TDM input, the ADAU1401A data starts after the edge defined by ILP. The ADAU1401A 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 ADAU1401A 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 Analog Devices codecs in auxiliary mode. To work in this mode with either the input or output serial ports, set the ADAU1401A 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.	
	Settings	**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

ADDRESS 2080 AND ADDRESS 2081 (0x0820 AND 0x0821)—MULTIPURPOSE PIN CONFIGURATION REGISTERS

Each multipurpose pin can be set to different functions from these registers (Address 2080 and Address 2081). The two 3-byte registers are broken up into 12 4-bit (nibble) sections that each control a different MP pin. Table 54 lists the function of each nibble setting within the MPx pin configuration registers. The MSB of each pin's 4-bit configuration inverts the input to or output from the pin. The internal pull-up resistor (approximately 15 kΩ) of each MPx pin is enabled when it is set as a digital input (either a GPIO input or a serial data port input).

Table 52. Register 2080 Bit Map

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
								MP53	MP52	MP51	MP50	MP43	MP42	MP41	MP40	0x00
MP33	MP32	MP31	MP30	MP23	MP22	MP21	MP20	MP13	MP12	MP11	MP10	MP03	MP02	MP01	MP00	0x0000

Table 53. Register 2081 Bit Map

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
								MP113	MP112	MP111	MP110	MP103	MP102	MP101	MP100	0x00
MP93	MP92	MP91	MP90	MP83	MP82	MP81	MP80	MP73	MP72	MP71	MP70	MP63	MP62	MP61	MP60	0x0000

Table 54. Multipurpose Pin Configuration Registers Bit Descriptions

Bit Name	Description	
MPx[3:0]	Set the function of each multipurpose pin.	
	Settings	**Function**
	1111	Auxiliary ADC input (see Table 63)
	1110	Reserved
	1101	Reserved
	1100	Serial data port—inverted (see Table 65)
	1011	Open-collector output—inverted
	1010	GPIO output—inverted
	1001	GPIO input, no debounce—inverted
	1000	GPIO input, debounce—inverted
	0111	N/A
	0110	Reserved
	0101	Reserved
	0100	Serial data port (see Table 65)
	0011	Open-collector output
	0010	GPIO output
	0001	GPIO input, no debounce
	0000	GPIO input, debounce

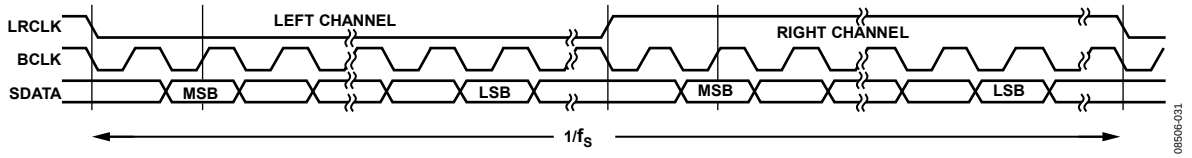

Figure 32. I²S Mode—16 Bits to 24 Bits per Channel

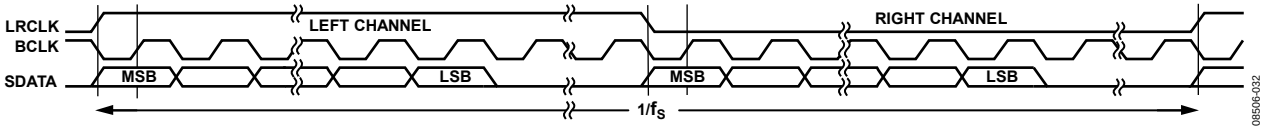

Figure 33. Left-Justified Mode—16 Bits to 24 Bits per Channel

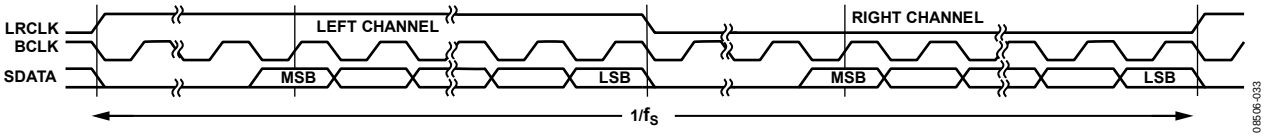

Figure 34. Right-Justified Mode—16 Bits to 24 Bits per Channel

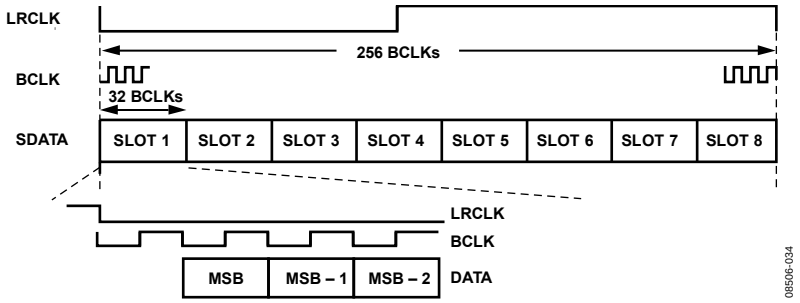

Figure 35. TDM Mode

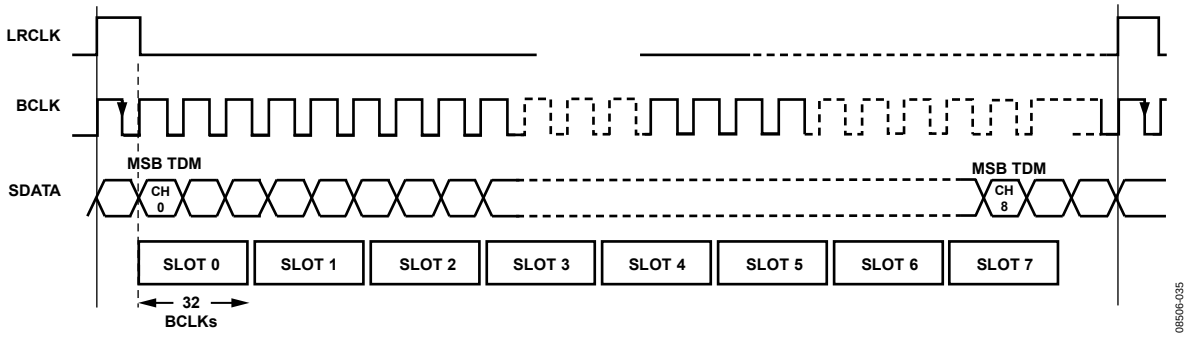

Figure 36. TDM Mode with Pulse Word Clock

TYPICAL APPLICATION SCHEMATICS

SELF-BOOT MODE

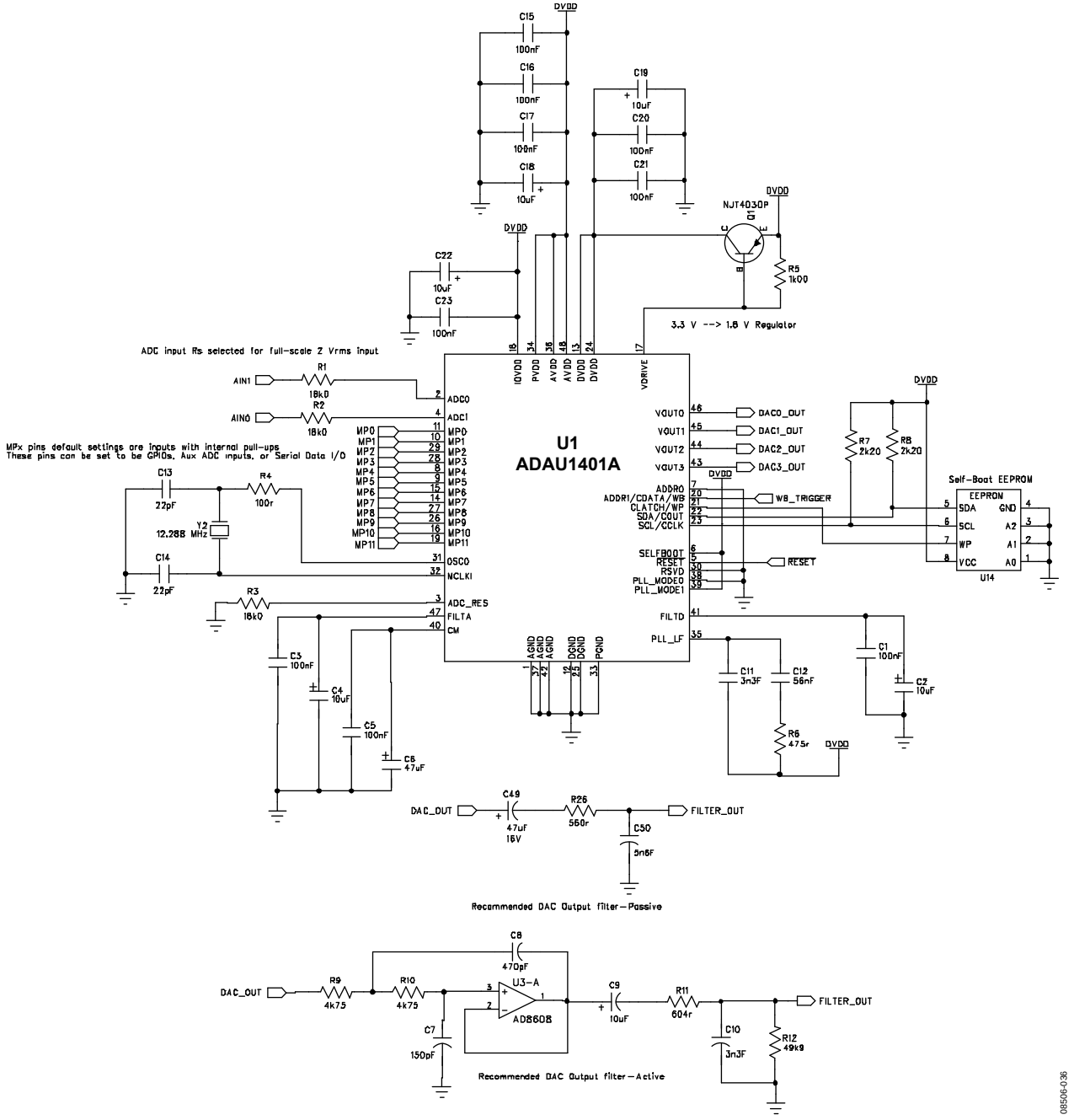

Figure 37. Self-Boot Mode Schematic

I²C CONTROL

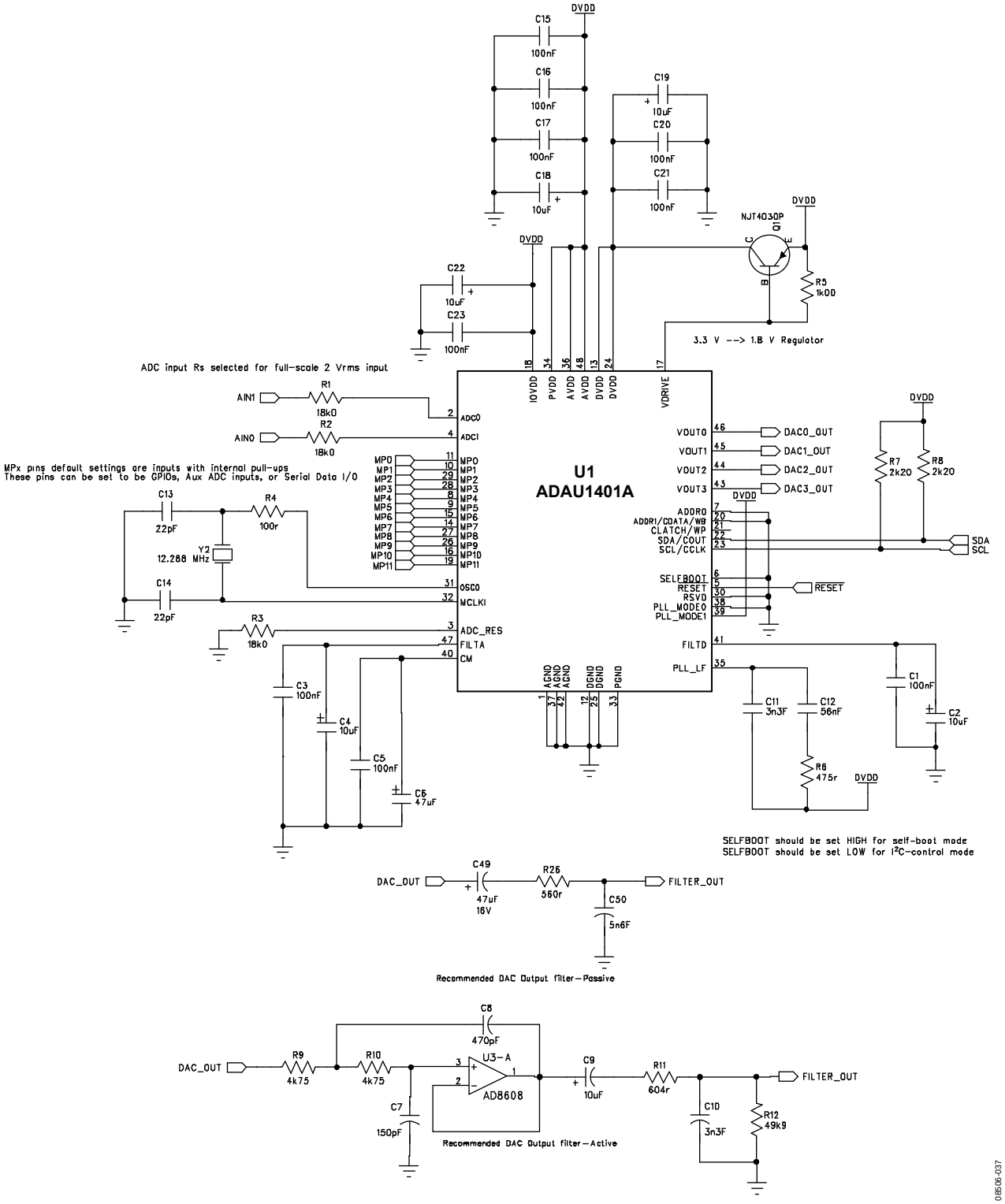

Figure 38. I²C Control Schematic

SPI CONTROL

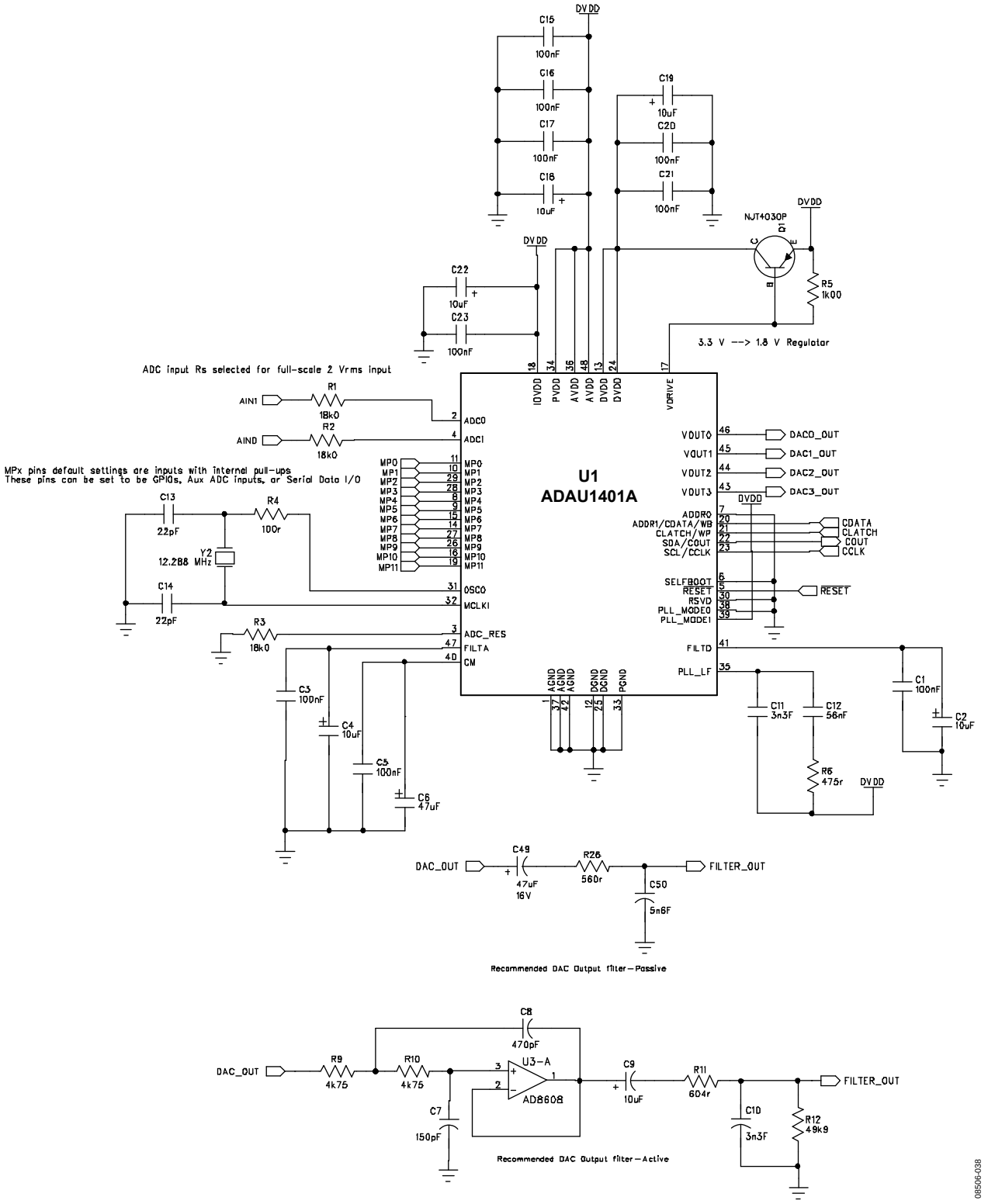

Figure 39. SPI Control Schematic

NOTES

I²C refers to a communications protocol originally developed by Philips Semiconductors (now NXP Semiconductors).

D08506-0-11/10(A)