Analog Interface ■ 8

8.1 OVERVIEW

The ADSP-21msp58 and ADSP-21msp59 processors include an analog signal interface consisting of a 16-bit sigma-delta A/D converter, a 16-bit sigma-delta D/A converter, and a set of memory-mapped control and data registers. The analog interface offers the following features:

- linear-coded 16-bit sigma-delta ADC
- linear-coded 16-bit sigma-delta DAC
- on-chip anti-aliasing and anti-imaging filters
- 8 kHz sampling frequency
- programmable gain for DAC and ADC
- on-chip voltage reference

The analog interface provides a complete analog front end for high performance voiceband DSP applications. The ADC and DAC operate at a fixed sampling rate of 8 kHz. The inclusion of on-chip anti-aliasing and anti-imaging filters, 16-bit sigma-delta converters, and programmable gain amplifiers ensures a highly integrated solution to voiceband analog processing requirements. Sigma-delta conversion technology eliminates the need for complex off-chip anti-aliasing filters and sample-and-hold circuitry.

The ADSP-21msp58 and ADSP-21msp59 contain the same analog interface—they differ only in the amount of on-chip memory. Refer to the *ADSP-21msp58/59 Data Sheet* for detailed analog performance specifications.

The analog interface of the ADSP-21msp58/59 is operated by using several data-memory-mapped control and data registers. The ADC and DAC I/O can be transmitted and received via individual memory-mapped registers, or the data can be autobuffered directly into the processor's data memory. This autobuffering is similar to serial port autobuffering, as described in Chapter 5.

8 Analog Interface

Two ADSP-21msp58/59 interrupts are dedicated to the ADC and DAC converters. One interrupt is used for the ADC and the other interrupt is used for the DAC. Interrupts occur at the sample rate or when the autobuffer transfer is complete.

A block diagram of the analog interface is shown in Figure 8.1, and pin definitions are given in Table 8.1.

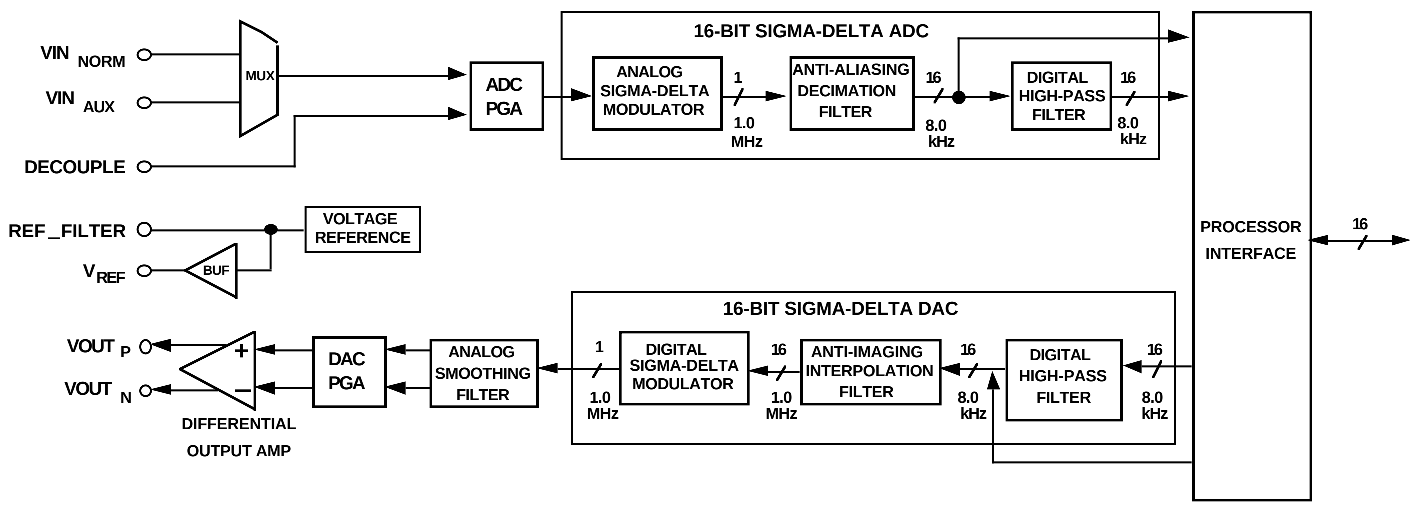


Figure 8.1 Analog Interface Block Diagram (ADSP-21msp58/59)

8.2 A/D CONVERSION

The A/D conversion circuitry of the ADSP-21msp58/59's analog interface consists of an input multiplexer, a programmable gain amplifier (PGA), and a sigma-delta analog-to-digital converter (ADC).

8.2.1 Analog Input

The analog input is internally biased by an on-chip voltage reference to allow operation of the ADSP-21msp58/59 with a single +5V power supply. The analog inputs should be ac-coupled.

An analog multiplexer selects either the NORM or AUX input. The input multiplexer is configured by bit 1 (IMS) of the ADSP-21msp58/59's analog control register (which is memory-mapped at address 0x3FEE in data memory). The multiplexer setting should not be changed while an input signal is being processed.

Pin Name	*I/O*	*Function*
VIN_{NORM}	I	Input terminal of the NORM channel of the ADC.
VIN_{AUX}	I	Input terminal of the AUX channel of the ADC.
Decouple	I	Ground reference of the NORM and AUX channels for the ADC.
$VOUT_{P}$	O	Non-inverting output terminal of the differential output amplifier from the DAC.
$VOUT_{N}$	O	Inverting output terminal of the differential output amplifier from the DAC.
V_{REF}	O	Buffered output voltage reference.
REF_FILTER	O	Voltage reference external bypass filter node.
V_{CC}		Analog supply voltage.
GND_{A}		Analog ground.

Table 8.1 Analog Interface Pin Definitions

The ADC PGA may be used to increase the signal level by +6 dB, +20 dB, or +26 dB. This selection is configured by bits 9 and 0 (IG1, IG0) of the analog control register. Input signal level to the sigma-delta modulator should not exceed the V_{INMAX} specification listed in the *ADSP-21msp58/59 Data Sheet*. Refer to "Analog Input" in the "Design Considerations" section of this chapter for more information.

An offset may be added to the input of the ADC in order to move the ADC's idle tones out of the 4.0 kHz speech band range. This is selected by bit 10 of the analog control register. The added offset must be removed by the ADC's high pass filter; therefore the high pass filter must be inserted (not bypassed) when the offset is added.

8.2.2 ADC

The analog interface's ADC consists of a 4th-order analog sigma-delta modulator, an anti-aliasing decimation filter, and a digital high pass filter. The sigma-delta modulator noise-shapes the signal and produces 1-bit samples at a 1.0 MHz rate. This bit stream, which represents the analog input signal, is fed to the anti-aliasing decimation filter.

8.2.2.1 Decimation Filter

The ADC's anti-aliasing decimation filter contains two stages. The first stage is a $sinc^4$ digital filter that increases resolution to 16 bits and reduces the sample rate to 40 kHz. The second stage is an IIR low pass filter.

The IIR low pass filter is a 10th-order elliptic filter with a passband edge at 3.7 kHz and a stopband attenuation of 65 dB at 4 kHz. This filter has the following specifications:

Filter type:	10th-order low pass elliptic IIR
Sample frequency:	40.0 kHz
Passband cutoff*:	3.70 kHz
Passband ripple:	±0.2 dB
Stopband cutoff:	4.0 kHz
Stopband ripple:	–65.00 dB

* The passband cutoff frequency is defined to be the last point in the passband that meets the passband ripple specification.

(Note that these specifications apply only to this filter, and not to the entire ADC. The specifications can be used to perform further analysis of the exact characteristics of the filter, for example using a digital filter design software package.)

Figure 8.2 shows the frequency response of the IIR low pass filter.

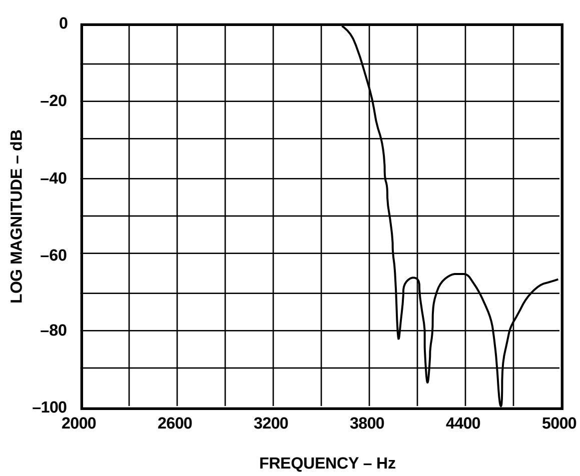


Figure 8.2 IIR Low Pass Filter Frequency Response

8.2.2.2 High Pass Filter

The ADC's digital high pass filter removes frequency components at the low end of the spectrum; it attenuates signal energy below the passband of the converter. The ADC's high pass filter can be bypassed by setting bit 7 (ADBY) of the ADSP-21msp58/59's analog control register.

The high pass filter is a 4th-order elliptic filter with a passband cutoff at 150 Hz. Stopband attenuation is 25 dB. This filter has the following specifications:

Filter type:	4th-order high pass elliptic IIR
Sample frequency:	8.0 kHz
Passband cutoff:	150.0 Hz
Passband ripple:	±0.2 dB
Stopband cutoff:	100.0 Hz
Stopband ripple:	–25.00 dB

(Note that these specifications apply only to this filter, and not to the entire ADC. The specifications can be used to perform further analysis of the exact characteristics of the filter, for example using a digital filter design software package.)

Figure 8.3 shows the frequency response of the high pass filter.

Passband ripple is ±0.2 dB for the combined effects of the ADC's digital filters (i.e. high pass filter and IIR low pass of the decimation filter) in the 300–3400 Hz passband.

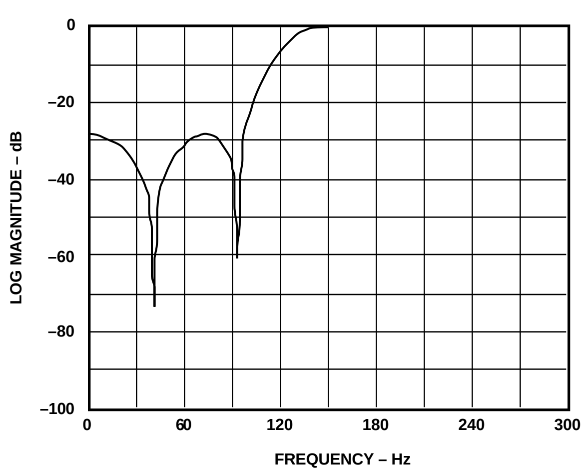


Figure 8.3 High Pass Filter Frequency Response

8.3 D/A CONVERSION

The D/A conversion circuitry of the ADSP-21msp58/59's analog interface consists of a sigma-delta digital-to-analog converter (DAC), an analog smoothing filter, a programmable gain amplifier, and a differential output amplifier.

8.3.1 DAC

The analog interface's DAC implements digital filters and a sigma-delta modulator with the same characteristics as the filters and modulator of the ADC. The DAC consists of a digital high pass filter, an anti-imaging interpolation filter, and a digital sigma-delta modulator.

The DAC receives 16-bit data values from the ADSP-21msp58/59's DAC Transmit data register (which is memory-mapped at address 0x3FEC in data memory). The data stream is filtered first by the DAC's high pass filter and then by the anti-imaging interpolation filter. These filters have the same characteristics as the ADC's anti-aliasing decimation filter and digital high pass filter.

The output of the interpolation filter is fed to the DAC's digital sigma-delta modulator, which converts the 16-bit data to 1-bit samples at a 1.0 MHz rate. The modulator noise-shapes the signal such that errors inherent to the process are minimized in the passband of the converter.

The bit stream output of the sigma-delta modulator is fed to the DAC's analog smoothing filter where it is converted to an analog voltage.

8.3.1.1 High Pass Filter

The DAC's digital high pass filter has the same characteristics as the high pass filter of the ADC. The high pass filter removes frequency components at the low end of the spectrum; it attenuates signal energy below the passband of the converter. The DAC's high pass filter can be bypassed by setting bit 8 (DABY) of the ADSP-21msp58/59's analog control register.

The high pass filter is a 4th-order elliptic filter with a passband cutoff at 150 Hz. Stopband attenuation is 25 dB. This filter has the following specifications:

Filter type:	4th-order high pass elliptic IIR
Sample frequency:	8.0 kHz
Passband cutoff:	150.0 Hz
Passband ripple:	±0.2 dB
Stopband cutoff:	100.0 Hz
Stopband ripple:	–25.00 dB

(Note that these specifications apply only to this filter, and not to the entire DAC. The specifications can be used to perform further analysis of the exact characteristics of the filter, for example using a digital filter design software package.)

Figure 8.3 shows the frequency response of the high pass filter.

8.3.1.2 Interpolation Filter

The DAC's anti-imaging interpolation filter contains two stages. The first stage is is an IIR low pass filter that interpolates the data rate from 8 kHz to 40 kHz and removes images produced by the interpolation process. The output of this stage is then interpolated to 1.0 MHz and fed to the second stage, a $sinc^4$ digital filter that attenuates images produced by the 40 kHz to 1.0 MHz interpolation process.

The IIR low pass filter is a 10th-order elliptic filter with a passband edge at 3.70 kHz and a stopband attenuation of 65 dB at 4 kHz. This filter has the following specifications:

Filter type:	10th-order low pass elliptic IIR
Sample frequency:	40.0 kHz
Passband cutoff*:	3.70 kHz
Passband ripple:	±0.2 dB
Stopband cutoff:	4.0 kHz
Stopband ripple:	–65.00 dB

* The passband cutoff frequency is defined to be the last point in the passband that meets the passband ripple specification. (Note that these specifications apply only to this filter, and not to the entire DAC. The specifications can be used to perform further analysis of the exact characteristics of the filter, for example using a digital filter design software package.)

Figure 8.2 shows the frequency response of the IIR low pass filter.

Passband ripple is ±0.2 dB for the combined effects of the DAC's digital filters (i.e. high pass filter and IIR low pass of the interpolation filter) in the 300–3400 Hz passband.

8.3.1.3 Analog Smoothing Filter & Programmable Gain Amplifier

The DAC's programmable gain amplifier (PGA) can be used to adjust the output signal level by –15 dB to +6 dB. This gain is selected by bits 2-4 (OG0, OG1, OG2) of the of the ADSP-21msp58/59's analog control register.

The DAC's analog smoothing filter consists of a 2nd-order Sallen-Key continuous-time filter and a 3rd-order switched capacitor filter. The Sallen-Key filter has a 3 dB point at approximately 25 kHz.

8.3.2 Differential Output Amplifier

The ADSP-21msp58/59's analog output signal ($VOUT_P$ – $VOUT_N$) is produced by a differential amplifier. The differential amplifier meets specifications for loads greater than 2 kΩ ($R_L \geq$ 2 kΩ) and has a maximum differential output voltage swing of ±3.156 V peak-to-peak (3.17 dBm0). The DAC will drive loads smaller than 2 kΩ, but with degraded performance.

The output signal is dc-biased to the on-chip voltage reference (V_{REF}) and can be ac-coupled directly to a load or dc-coupled to an external amplifier. Refer to "Analog Output" in the "Design Considerations" section of this chapter for more information.

The $VOUT_P$ – $VOUT_N$ outputs must be used as a differential signal, otherwise performance will be severely degraded. Do not use either pin as a single-ended output.

8.4 OPERATING THE ANALOG INTERFACE

The analog interface of the ADSP-21msp58/59 is operated with the use of several memory-mapped control and data registers. The ADC and DAC I/O data can be received and transmitted in two memory-mapped data registers. The data can also be autobuffered into (and from) on-chip memory where data is automatically transferred to or from the data registers. In both cases, the I/O processing is interrupt-driven: two ADSP-21msp58/59 interrupts are dedicated to the analog interface, one for ADC receive data and one for DAC transmit data. (**Note:** Autobuffering with SPORT1 is not available on the ADSP-21msp5x processors because this autobuffering channel is used for the analog interface.)

The ADSP-21msp58/59 must have an input clock frequency of 13 MHz. At this frequency, analog-to-digital and digital-to-analog converted data is transmitted at an 8 kHz rate with a single 16-bit word transmitted every 125 µs.

8.4.1 Memory-Mapped Control Registers

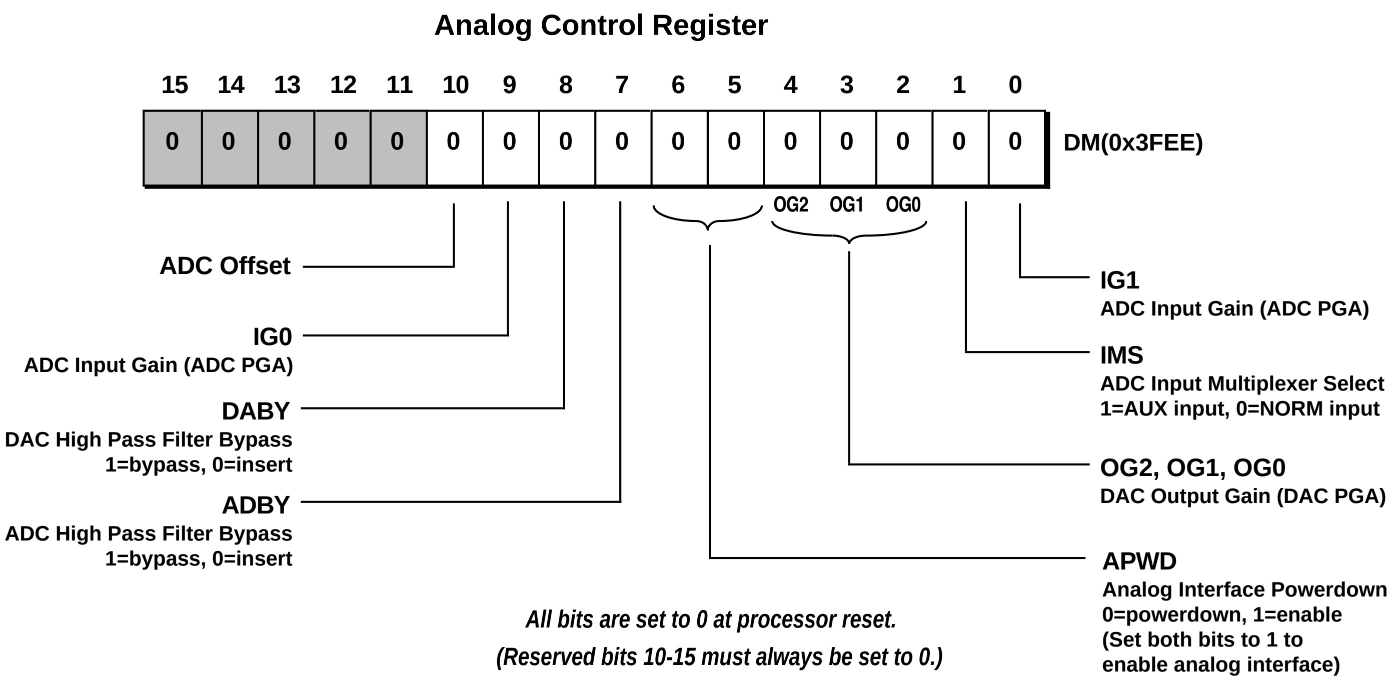
Two memory-mapped control registers are used to configure the ADSP-21msp58/59's analog interface: the analog control register and analog autobuffer/powerdown register.

8.4.1.1 Analog Control Register

The analog control register (located at address 0x3FEE in data memory) is shown in Figure 8.4. This register configures the ADC input multiplexer, ADC input gain PGA, ADC high pass filter, DAC high pass filter, and DAC output gain PGA.

The analog control register also contains the APWD bits (bits 5, 6) which must both be set to ones to enable and start up the analog interface—*always enable and disable the analog interface using both bits 5 and 6.* The DAC and ADC begin transmitting data after these bits are set. Clearing the APWD bits disables the entire analog interface by putting it in a powerdown state. The APWD bits must be cleared (to zeros) at least three processor cycles before putting the processor in powerdown. See "Powerdown" in Chapter 9, System Interface.

The analog control register is cleared (to 0x0000) by the processor's $\overline{RESET}$ signal. Note that bits 10-15 of this register are reserved and must always be set to zero.



IG1, IG0
ADC Input Gain (ADC PGA)

Gain	IG1	IG0
0 dB	0	0
+6 dB	0	1
+20 dB	1	0
+26 dB	1	1

OG2, OG1, OG0
DAC Output Gain (DAC PGA)

Gain	OG2	OG1	OG0
+6 dB	0	0	0
+3 dB	0	0	1
0 dB	0	1	0
–3 dB	0	1	1
–6 dB	1	0	0
–9 dB	1	0	1
–12 dB	1	1	0
–15 dB	1	1	1

Figure 8.4 Analog Control Register

8.4.1.2 Analog Autobuffer/Powerdown Register

The analog autobuffer/powerdown register (located at address 0x3FEF in data memory) is shown in Figure 8.5. This register enables or disables autobuffering of ADC receive data and/or DAC transmit data—autobuffering is enabled by writing ones to the ARBUF (bit 0) and/or ATBUF (bit 1) bits. When autobuffering is enabled, I (index) and M (modify) registers are selected in bits 2–11 for the receive and/or transmit data buffers. See “Autobuffering” in the Serial Ports chapter for details on autobuffering.

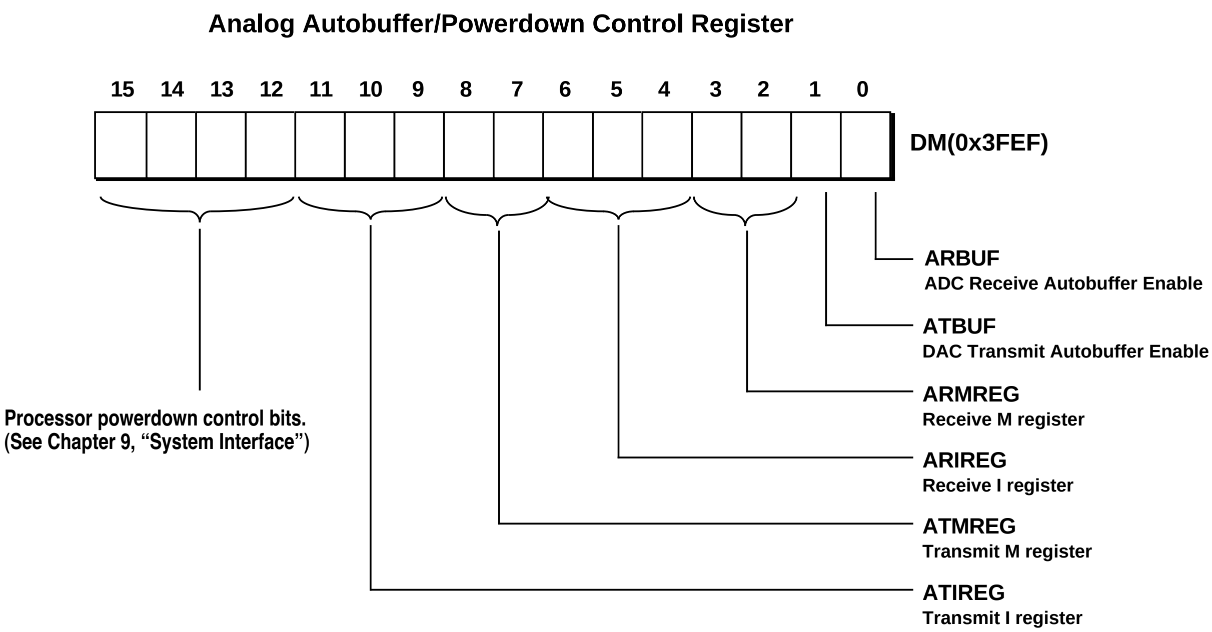


Figure 8.5 Analog Autobuffer/Powerdown Control Register

Bits 12–15 of the analog autobuffer/powerdown register control the ADSP-21msp58/59's processor powerdown function, *not* powerdown of the analog interface—powerdown of the analog interface only is controlled by the APWD bits (bits 5, 6) of the analog control register. The ADSP-21msp58/59's powerdown function is described in the "Powerdown" section of Chapter 9, System Interface.

8.4.2 Memory-Mapped Data Registers

There are two memory-mapped data registers dedicated to the analog interface. The 16-bit ADC receive data register is located at address 0x3FED in data memory. The 16-bit DAC transmit data register is located at address 0x3FEC in data memory. These registers must be individually read and written when autobuffering is not in use (autobuffering automatically transfers the data to and from processor data memory).

When autobuffering is disabled, data must be transmitted to the sigma-delta DAC by writing a 16-bit word to the DAC transmit register (0x3FEC) and data must be received from the sigma-delta ADC by reading a 16-bit word from the ADC receive register (0x3FED).

8.4.3 ADC & DAC Interrupts

The analog interface generates two interrupts that signal either:
1) that a 16-bit, 8 kHz analog-to-digital or digital-to-analog conversion has been completed, or 2) that an autobuffer block transfer has been completed (i.e. the entire data buffer contents have been transmitted or received).

When one of the analog interrupts occurs, the processor vectors to the appropriate address:

DAC Transmit interrupt vector address: 0x18
ADC Receive interrupt vector address: 0x1C

These interrupts can be masked out in the processor's IMASK register and can be forced or cleared in the IFC register.

8.4.3.1 Autobuffering Disabled

The ADC receive and DAC transmit interrupts occur at an 8 kHz rate, indicating when the data registers should be accessed, when autobuffering is disabled. On the receive side, the ADC interrupt is generated each time an A/D conversion cycle is completed and the 16-bit data word is available in the ADC receive register. On the transmit side, the DAC interrupt is generated each time a D/A conversion cycle is completed and the DAC transmit register is ready for the next 16-bit data word.

Both interrupts are generated simultaneously at an 8 kHz rate, occurring every 3250 instruction cycles with a 13 MHz internal clock, when autobuffering is disabled. The interrupts are generated continuously, starting when the analog interface is powered up by setting the APWD bits (bits 5, 6) to ones in the analog control register. Because both interrupts occur simultaneously, only one should be enabled (in IMASK) to vector to a single service routine that handles both transmit and receive data. (When autobuffering is enabled, though, both interrupts should be enabled.)

A simple analog loopback program is shown in Listing 8.1.

```
{ ADSP-21msp58/59 Analog Interface Loopback Example   }
{   - configures analog interface                     }
{   - copies ADC receive data to DAC transmit buffer}


.MODULE/ABS=0/BOOT=0  talkthru;

#define  codec_tx_data  0x3FEC
#define  codec_rx_data  0x3FED
#define  codec_ctrl_reg 0x3FEE

resetv:         JUMP setup; NOP; NOP; NOP;
irq2v:          RTI; NOP; NOP; NOP;         {interrupt vectors ...}
hipwv:          RTI; NOP; NOP; NOP;
hiprv:          RTI; NOP; NOP; NOP;
spt0tv:         RTI; NOP; NOP; NOP;
spt0rv:         RTI; NOP; NOP; NOP;
antv:           RTI; NOP; NOP; NOP;
anrv:           SI = DM(codec_rx_data);     {read in data from ADC}
                DM(codec_tx_data) = SI;     {write out data to DAC}
                RTI; NOP;
irq1v:          RTI; NOP; NOP; NOP;
irq0v:          RTI; NOP; NOP; NOP;
timerv:         RTI; NOP; NOP; NOP;
pwrdwnv:        RTI; NOP; NOP; NOP;

setup:          AX1 = 0x0060;
                DM(codec_ctrl_reg) = AX1;   {power up analog interface}
                IMASK = 0x8;                {enable analog receive interrupt}
wait_loop:      IDLE;                       {wait for interrupt}
                JUMP wait_loop;
.ENDMOD;
```

Listing 8.1 ADSP-21msp58/59 Analog Loopback Program

8.4.3.2 Autobuffering Enabled

In some applications it is advantageous to perform block data transfers between the analog converters and processor memory. Analog interface autobuffering allows you to automatically transfer blocks of data from the ADC to on-chip processor data memory or from on-chip processor data memory to the DAC.

An interrupt is generated when an entire block transfer is complete (i.e. when the data buffer is full or empty). Analog interface autobuffering operates in the same way as SPORT autobuffering, described in Chapter 5. Note that data can be autobuffered through the analog converters or through SPORT0 of the ADSP-21msp58/59. Autobuffering is not available on SPORT1 of the ADSP-21msp58/59.

Before autobuffering is enabled, separate circular buffers must be set up in data memory for the ADC receive and DAC transmit data. This is accomplished by selecting I (index) and M (modify) registers in the analog autobuffer/powerdown register; see Figure 8.5.

Transmit data autobuffered to the DAC is addressed with the I register specified in the ATIREG field (bits 9, 10, 11). Receive data autobuffered from the ADC is addressed with the I register specified in the ARIREG field (bits 4, 5, 6). The modify (M) registers are specified in the ARMREG (bits 2, 3) field and ATMREG (bits 7, 8) field. Since the transfer of ADC and DAC data occurs simultaneously, it is possible to use the same I register for transmit and receive autobuffering. In this case, the buffer is shared for both functions and care should be taken when specifiying a value for the M register.

An autobuffering example program is shown in Listing 8.2.

```
{ ADSP-21msp58/59 Analog Interface Autobuffer Example }
{   - configures analog interface                     }
{   - enables analog autobuffer                       }
{   - receive analog data into a 256 word buffer      }
{   - transmit analog data from a 256 word buffer     }

.MODULE/RAM/ABS=0/BOOT=0  auto_example;
.VAR/DM/CIRC  buff1[256];                    {first data buffer}
.VAR/DM/CIRC  buff2[256];                    {second data buffer}
.VAR/DM  flag_bit;                           {tracks buffers}
#define  codec_tx_data  0x3FEC
#define  codec_rx_data  0x3FED
#define  codec_ctrl_reg 0x3FEE
#define codec_auto_ctrl 0x3FEF

resetv:        JUMP setup; NOP; NOP; NOP;
irq2v:         RTI; NOP; NOP; NOP;            {interrupt vectors ...}
hipwv:         RTI; NOP; NOP; NOP;
hiprv:         RTI; NOP; NOP; NOP;
spt0tv:        RTI; NOP; NOP; NOP;
spt0rv:        RTI; NOP; NOP; NOP;
antv:          RTI; NOP; NOP; NOP;
anrv:          JUMP switch; NOP; NOP; NOP;    {call autobuffer switch}
irq1v:         RTI; NOP; NOP; NOP;
irq0v:         RTI; NOP; NOP; NOP;
timerv:        RTI; NOP; NOP; NOP;
pwrdwnv:       RTI; NOP; NOP; NOP;
```

```
setup:          I0 = ^buff1;                {I0 points to first data buffer}
                L0 = %buff1;
                I1 = ^buff2;                {I1 points to second data buffer}
                L1 = %buff2;
                M0 = 0x1;
                SI = 0x0;                   {initialize flag register}
                DM(flag_bit) = SI;
                                            {use I1 and M0 for tranmsit}
                                            {use I0 and M0 for receive}
                AY0 = 0x0203;               {enable rcv and tx autobuffer}
                DM(codec_auto_ctrl) = AY0;
                AX1 = 0x0060;
                DM(codec_ctrl_reg) = AX1;   {power up analog interface}
                IMASK = 0x8;                {enable analog rx interrupt}

wait:           IDLE;                       {wait for autobuffer interrupt}
                JUMP wait;
switch:         AX0 = DM(flag_bit);
                AR = pass AX0;              {check buffer status}
                IF NE JUMP fill_buff2;
fill_buff1:     SI = 0x1;                   {fill buff2 next time}
                AY0 = 0x0013;
                JUMP done;
fill_buff2:     SI = 0x0;                   {fill buff1 next time}
                AY0 = 0x0203;
                JUMP done;
done:           DM(codec_auto_ctrl) = AY0;
                DM(flag_bit) = SI;
                RTI;
.ENDMOD;
```

Listing 8.2 ADSP-21msp58/59 Analog Autobuffer Program

Receive and transmit autobuffering may be independently enabled and the two interrupts can occur (and be serviced) independently. This allows the use of different data buffer lengths when autobuffering both receive and transmit data. It also allows autobuffering to be used on only one side, receive or transmit, while the other is serviced at the 8 kHz interrupt rate.

8.5 CIRCUIT DESIGN CONSIDERATIONS

The following sections discuss interfacing analog signals to the ADSP-21msp58/59.

8.5.1 Analog Signal Input

Figure 8.6 shows the recommended input circuit for the ADSP-21msp58/59's analog input pin (either VIN_{NORM} or VIN_{AUX}). The circuit of Figure 8.6 implements a first-order low pass filter (R_1C_1). The 3 dB point of the filter should be less than 40 kHz. This is the only filter that must be implemented external to the processor to prevent aliasing of the sampled signal. Since the ADSP-21msp58/59's sigma-delta ADC uses a highly oversampled approach that transfers most of the anti-aliasing filtering into the digital domain, the off-chip anti-aliasing filter need only be of low order. Refer to the *ADSP-21msp58/59 Data Sheet* for more detailed information.

The ADSP-21msp58/59's on-chip ADC PGA (programmable gain amplifier) can be used when there is not enough gain in the input circuit. The ADC PGA is configured by bits 9 and 0 (IG1, IG0) of the processor's analog control register. The gain must be selected to ensure that a full-scale input signal (at R_1 in Figure 8.6) produces a signal level at the input to the sigma-delta modulator of the ADC that does not exceed V_{INMAX} (which is specified in the data sheet).

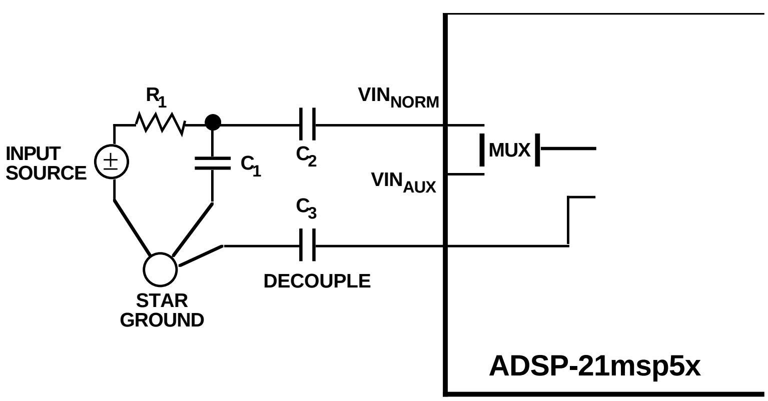


Figure 8.6 Recommended Analog Input Circuit

VIN_{NORM} and VIN_{AUX} are biased at the internal voltage reference (nominally 2.5V) of the ADSP-21msp58/59, which allows the analog interface to operate from a single supply. The input signal should be ac-coupled with an external capacitor (C_2). The value of C_2 is determined by the input resistance of the analog input (VIN_{NORM} , VIN_{AUX}), 200 kΩ, and the desired cutoff frequency. The cutoff frequency should be less than or equal to 30 Hz. The following equations should be used to determine the values for R_1, C_1, and C_2: R_1 should be less than or equal to 2.2 kΩ, C_2 should be greater than or equal to 0.027 µF, C_3 should be equal to C_2.

$$C_2 = \frac{1}{2\pi f_1 R_{IN}}$$

R_{IN} = input resistance of ADSP-21msp58/59 (200 kΩ)

f_1=cutoff frequency ≤ 30 Hz

$$R_1 = \frac{1}{2\pi f_2 C_2}$$

$R_1 \leq$ 2.2 kΩ

20 kHz < f_2 < 40 kHz *

$$C_1 = \frac{1}{2\pi f_2 R_1}$$

$C_3 = C_2$

* If minimum (< 0.1 dB) rolloff at 4 kHz is desired, f_2 should be set to 40 kHz.

8.5.2 Analog Signal Output

The ADSP-21msp58/59's differential analog output ($VOUT_P$ – $VOUT_N$) is produced by an on-chip differential amplifier. The differential amplifier will meet dynamic specifications for loads greater than 2 kΩ ($R_L \geq$ 2 kΩ) and has a maximum differential output voltage swing of ±3.156 V peak-to-peak (3.17 dBm0). The DAC will drive loads smaller than 2 kΩ, but with degraded dynamic performance. The differential output can can be ac-coupled directly to a load or dc-coupled to an external amplifier.

Figure 8.7 shows a simple circuit providing a differential output with ac coupling. The capacitor of this circuit (C_{OUT}) is optional; if used, its value can be chosen as follows:

$$C_{OUT} = \frac{1}{60\pi R_L}$$

The $VOUT_P$ – $VOUT_N$ outputs must be used as differential outputs; do not use either as a single-ended output. Figure 8.8 shows an example circuit which can be used to convert the differential output to a single-ended output. The circuit uses a differential-to-single-ended amplifier, the Analog Devices SSM-2141.

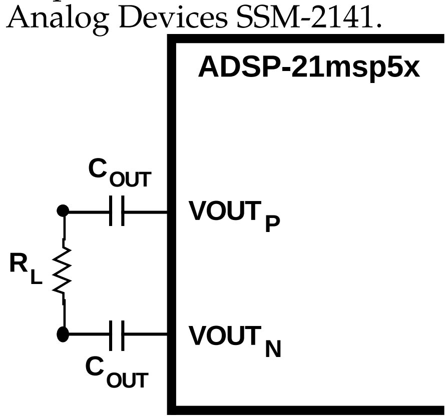


Figure 8.7 Example Circuit For Differential Output With AC Coupling

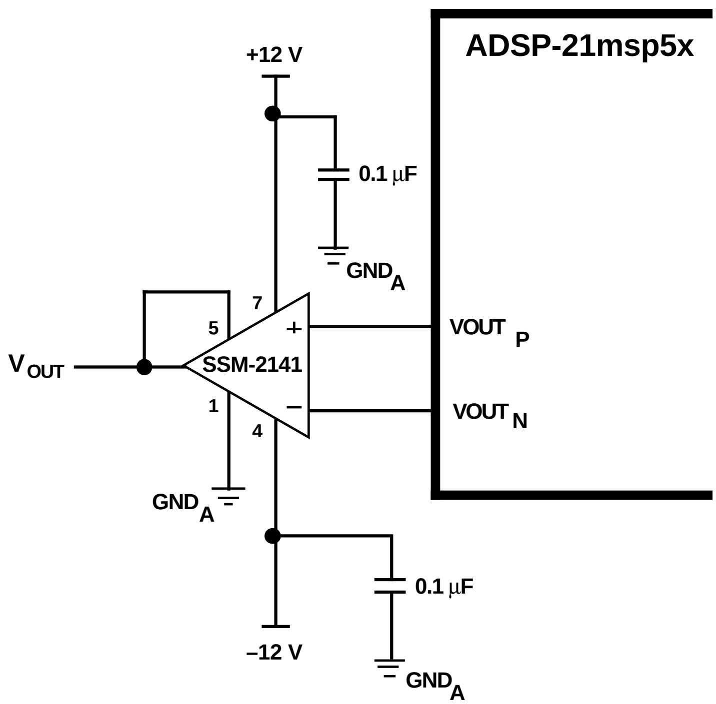


Figure 8.8 Example Circuit For Single-Ended Output

8.5.3 Voltage Reference Filter Capacitance

Figure 8.9 shows the recommended reference filter capacitor connections. The capacitor grounds should be connected to the same star ground point as that of Figure 8.6.

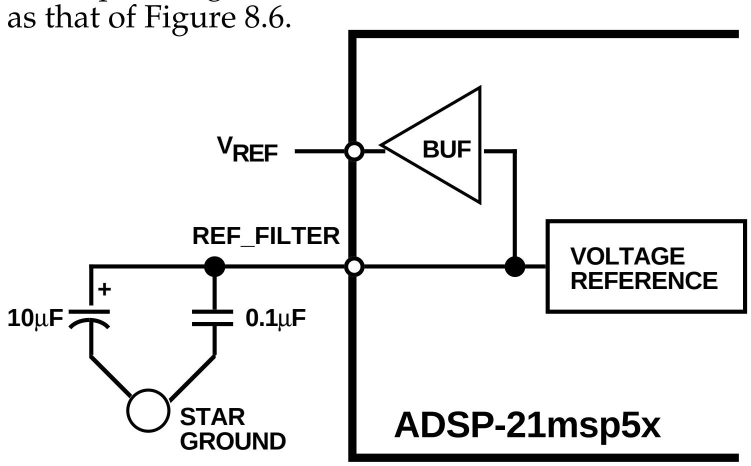


Figure 8.9 Voltage Reference Filter Capacitor