ANALOG DEVICES

AN-406
APPLICATION NOTE

ONE TECHNOLOGY WAY • P.O. BOX 9106 • NORWOOD, MASSACHUSETTS 02062-9106 • 617/329-4700

Using the AD771x Family of 24-Bit Sigma-Delta A/D Converters

by Eamon Nash

INTRODUCTION

The AD771x Series is a family of ADCs that are designed specifically for low frequency, high accuracy industrial transducer applications. Figure 1 shows a block diagram of one of these devices. The analog front end, which in some cases also includes excitation sources for sensors, accepts low level signals that can be amplified by the internal programmable gain amplifier before being applied to a sigma-delta modulator. The pulse stream appearing at the output of this modulator is applied to a digital filter which produces a low-pass filtered, high resolution, serial output.

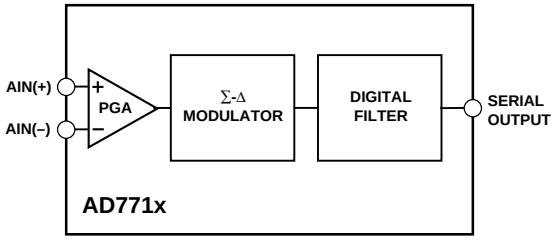


Figure 1. AD771x Block Diagram

TYPICAL APPLICATIONS

These devices are most often used to measure signals from thermocouples, resistive bridges and resistance temperature detectors (RTDs). These sensors, which output low level signals, generally measure slowly changing physical properties such as temperature, pressure or weight. Due to the frequent presence of common-mode voltages, differential signal measurement is desirable. If the sensor is located remotely, connection to the ADC is often over a long transmission line which can be susceptible to electromagnetically induced interference (EMI).

Figure 2 shows the AD7710 used in a thermocouple application. The sensor is connected to AIN1 which includes a burn-out detector current source. When this 100 nA current source is switched on, the AIN(+) pin will be driven to saturation if the thermocouple is open circuit.

While it is possible to use an analog technique to adjust for the cold junction temperature, it may be simpler to measure the temperature at the cold junction and subtract this from the thermocouple reading in the digital domain. The thermistor in Figure 2 which measures the cold junction temperature is excited by an onboard 20 μA current source.

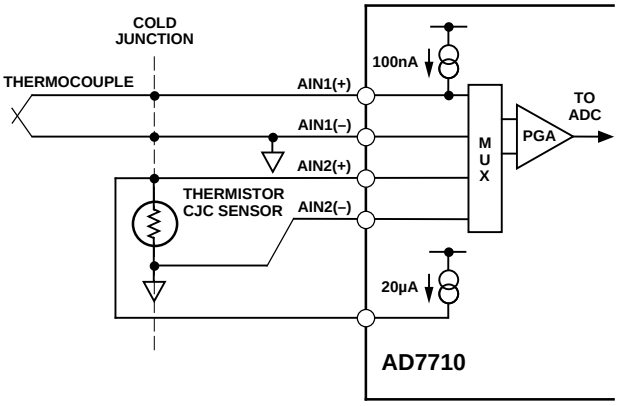


Figure 2. Thermocouple Interface

The thermocouple is grounded at AIN1(–) in order to provide a return path for the input leakage currents of the differential inputs that must flow at all times. If the voltage coming from the thermocouple has a common-mode voltage component, this short may be replaced by a resistor to ground. Due to the Johnson noise which this resistor generates, it should not be any larger than about 100 kΩ. In applications where the thermocouple is electrically connected to a chassis whose ground is referred to the local ground, this connection is unnecessary.

INPUT FILTERING CONSIDERATIONS

Before looking at some filtering schemes, it is important to consider the internal filtering capabilities of the converter itself as well as the nature of the noise that is being filtered.

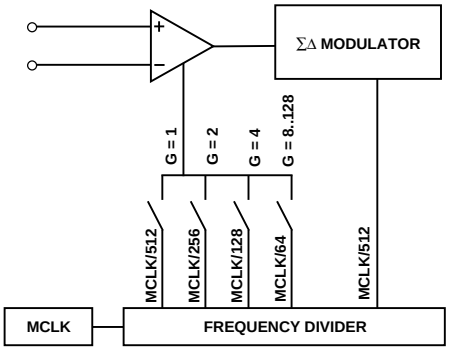
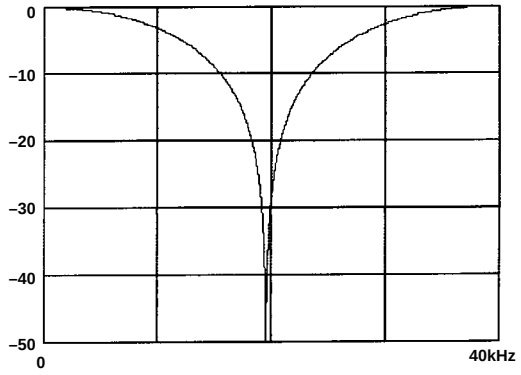


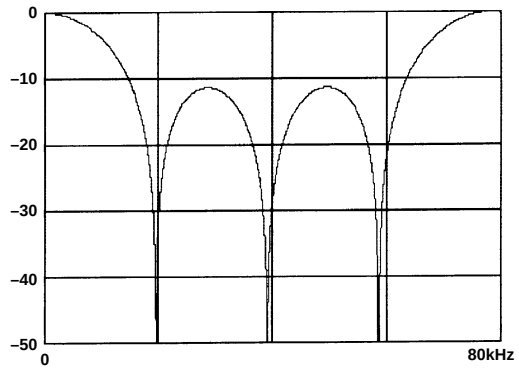
Figure 3. Input Sampling

As we can see from Figure 3, the sigma-delta modulator of the AD7710 operates at a frequency of $f_{CLK\ IN}/512$ (19.5 kHz @ $f_{CLK\ IN}$ = 10 MHz) which is independent of the gain of the programmable gain amplifier (PGA). The PGA itself, however, samples the input signal at a rate which increases as the programmed gain increases, from 19.5 kHz (MCLK/512) at a gain of 1 up to a maximum of 156 kHz (MCLK/64) for gains 8 through 128. At a gain of 8, for example, the input signal is sampled 8 times per modulator cycle. The average of the resulting samples is stored as a charge on a capacitor. Because the AIN(+) and AIN(–) inputs are sampled alternately and not simultaneously and because their samples are stored on the same capacitor, the input signal is actually being sampled at twice this rate (e.g., 16 times per modulator cycle at a gain of 8) which results in an overall sampling rate of 312 kHz at a gain of 8 (or 39 kHz at a gain of 1). This averaging causes input signals to be subjected to a low pass frequency response whose shape will depend upon the programmed gain. Figure 4 shows these frequency responses for the various programmed gains. Each response repeats itself at multiples of the effective sampling rate at that particular gain.

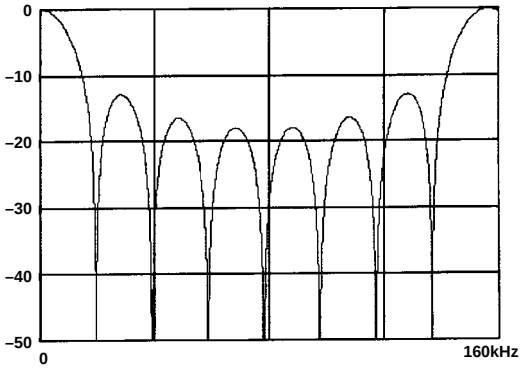
The frequency response of the digital filter in the output stage of the converter also repeats itself, but at multiples of the modulator frequency (19.5 kHz for the AD7710). Figure 5 shows this frequency response for the frequency ranges 0 kHz to 40 kHz, 0 kHz to 80 kHz, 0 kHz to 160 kHz and 0 kHz to 320 kHz. Combining these plots with those of Figure 4 gives the overall frequency response of the device, shown in Figure 6.



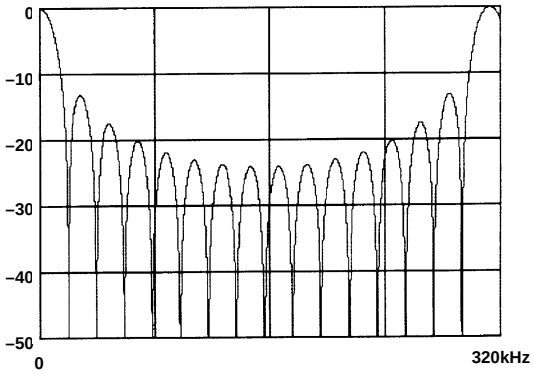
a. Gain = 1



b. Gain = 2

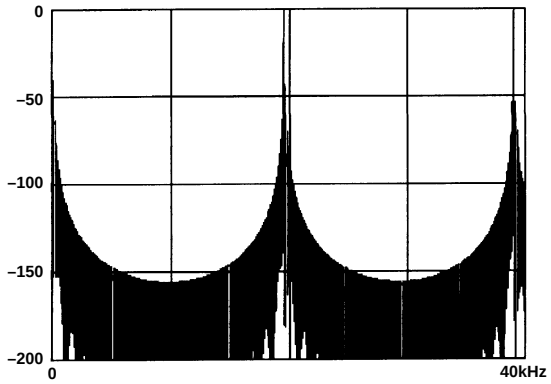


c. Gain = 4

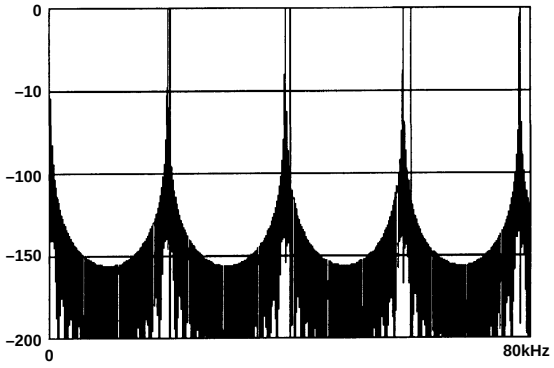


d. Gain = 8–128

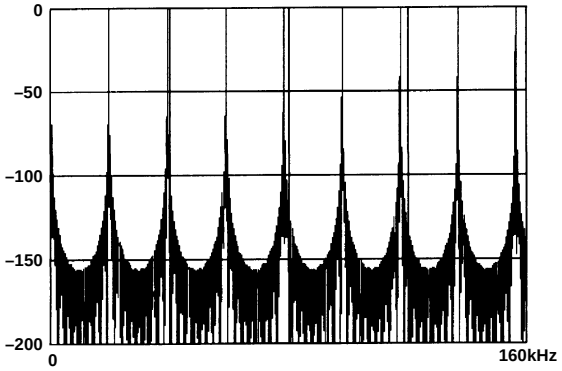
Figure 4. Frequency Response of PGA



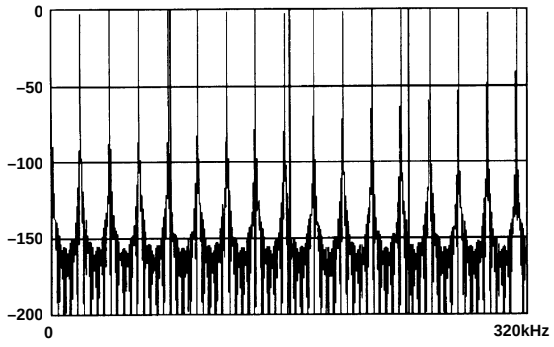
a. 0–40 kHz



b. 0–80 kHz

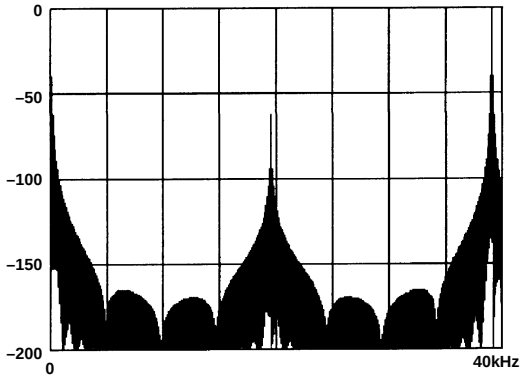


c. 0–160 kHz

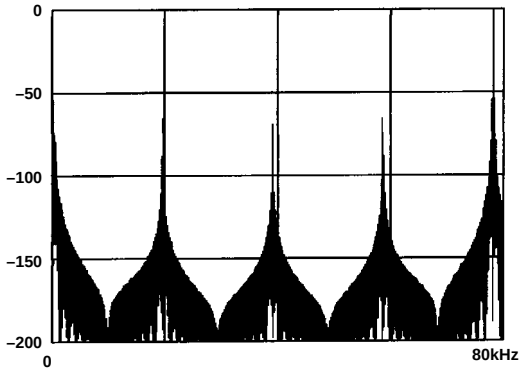


d. 0–320 kHz

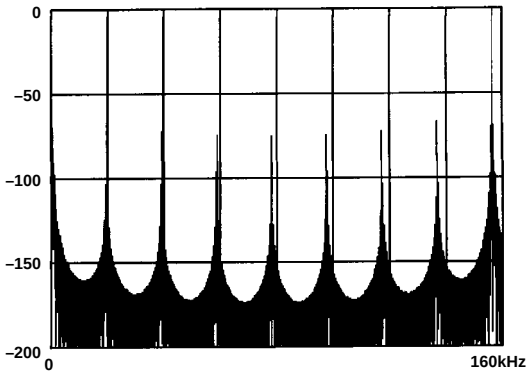
Figure 5. Frequency Response of Digital Filter



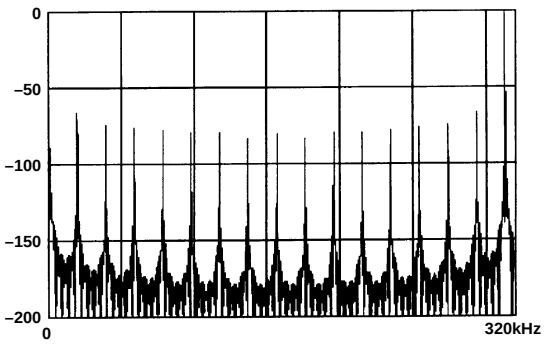
a. Gain = 1



b. Gain = 2

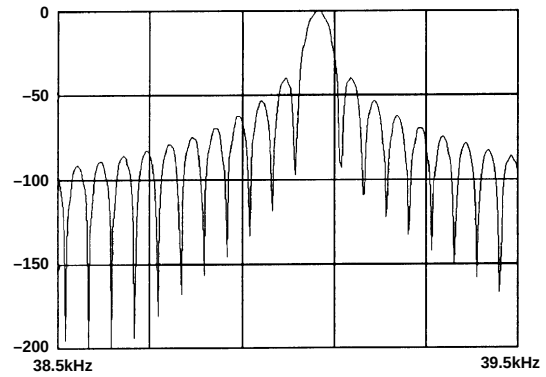


c. Gain = 4

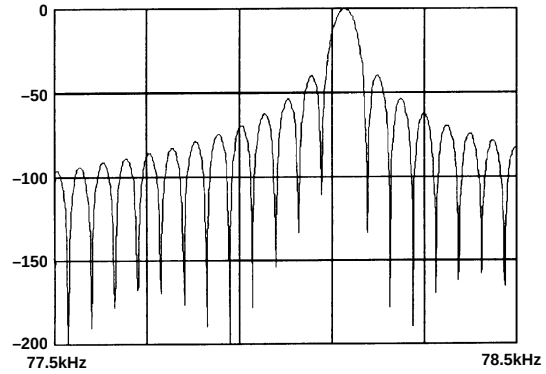


d. Gain = 8–128

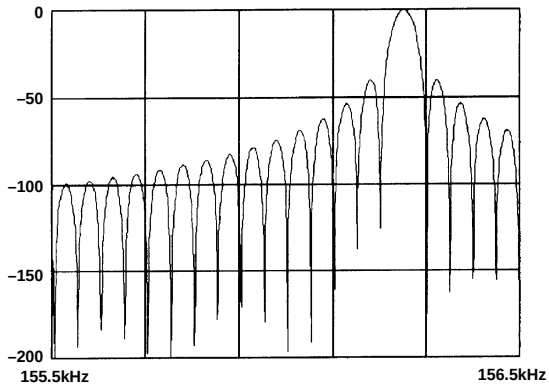
Figure 6. Overall Frequency Response of ADC



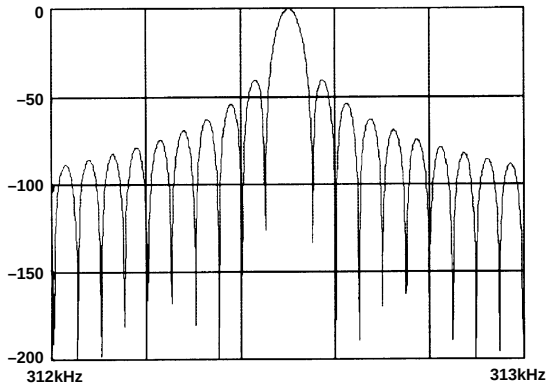
a. Gain = 1



b. Gain = 2



c. Gain = 4



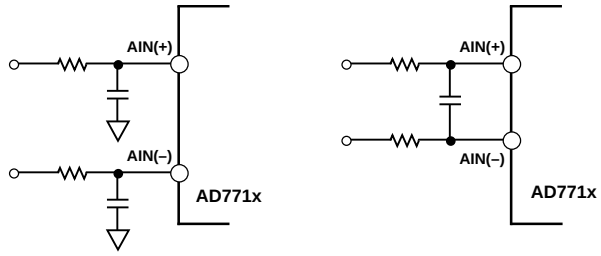
d. Gain = 8–128

Figure 7. Detail at 0 dB Passband

From Figure 6 it is clear that the device performs significant low-pass filtering on its input signal, the exact filter function being dependent upon the programmed gain. In each case the plot repeats itself at multiples of effective input sampling rate. At a gain of 2, for example, there is a narrow 0 dB passband at about 78 kHz as shown in Figure 6b. Noise in this frequency range will cause aliasing. However at all other frequencies (excluding multiples of 78 kHz), the device itself will supply sufficient filtering to prevent aliasing.

At gains from 8 to 128, the filtering effect is even more dramatic (Figure 6d). In this case input signals are attenuated by 60 dB or better up to a frequency of 312 kHz where there is a narrow 0 dB passband (Figure 7d).

Where external filtering is used, using two R-C networks is recommended as shown in Figure 8a. This will have good rejection of both differential noise and common-mode noise. Use of a differential filter as shown in Figure 8b will result in no rejection of common-mode noise and a maximum attenuation of –6 dB for differential mode interference.



a. *b.*

Figure 8. External Passive Filter Schemes

When using external filtering, it is important to remember the restrictions on the sizes of the passive components used. This subject is discussed in detail in the section “Antialias Considerations” in the AD771x data sheets. Where large RC values are required, either use active filtering to reduce source impedance or choose the AD7714 which contains a high impedance input buffer.

POST-FILTERING

It is also possible to reduce noise by post-filtering the output data from the device. One simple way of doing this is to implement a moving average digital filter in the microcontroller or microprocessor/DSP which reads the conversion results. For example, a 4-tap moving average filter would calculate the average of the last four conversions, after each conversion. While this has the advantage that the output data rate from the moving average filter is the same as that of the ADC, the step response of the overall system will be adversely affected.

The frequency response of a moving average filter has exactly the same form as the internal programmable gain amplifier of the ADC which is shown in Figure 4. Figure 4b shows the frequency response of the previously discussed 4-tap filter (Figure 4a represents a 2-tap filter, Figure 4c and 3d show an 8-tap and a 16-tap filter respectively). In this case the frequency response will repeat itself at multiples of the output data rate (this 0 dB point will however coincide with a notch in the internal digital filter of the ADC because the frequency of the first notch is also equal to the output data rate). So, for example, at an output data rate of 10 Hz, a 16-tap moving average filter will attenuate the frequency band 1 Hz to 9 Hz (and 11 Hz to 19 Hz and so on) by between 13 dB and 23 dB.

INPUT PREAMPLIFICATION

If the input signal to the converter needs to be preamplified and/or buffered, a single stage differential amplifier may be implemented using two operational amplifiers as shown in Figure 9.

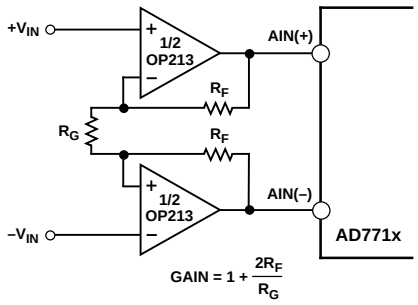


Figure 9. External Amplification

In choosing a suitable operational amplifier, low frequency noise (0.1 Hz to 10 Hz) is probably the most critical factor. The Referred To Input (RTI) device noise within the passband of the digital filter should be well below the LSB size that is being resolved. As the gain of the internal PGA is increased, this becomes even more critical. If, for example, the internal gain of the AD7711 is set to 128 in bipolar mode with a first notch frequency of 10 Hz (3 dB cutoff frequency = 2.62 Hz), 1 LSB in a 14-bit system is equal to 2.3 μV (5 V/128/2^{14}). If R_G and R_F in Figure 9 are set to the same value, the preamplifier gain of 3 reduces the RTI LSB size to about 800 nV. The OP213, which is used in Figure 9, has a low frequency (0.1 Hz to 10 Hz) noise level of 120 nV p-p. Quadratically summing the noise voltages from the two devices yields a total RTI noise level of about 170 nV p-p. At a notch frequency of 10 Hz, the –3 dB cutoff of the digital filter is at 2.62 Hz. This band-limiting reduces the RTI amplifier noise to about 70 nV which is well below the LSB size being resolved (800 nV).

INTERNAL GAIN VS. EXTERNAL GAIN

Where extremely high resolution is required, better noise performance may be achieved by setting the gain of the internal PGA to unity and externally amplifying the input signal using a very low noise operational amplifier such as the AD797. At a gain of 128 and at a notch frequency of 10 Hz, the AD7711 has an effective resolution of 16.5 bits which is derived from the RMS noise of the device (see Tables I and II of AD7711 data sheet). The number of noise free codes is calculated by converting this value from rms to p-p which involves dividing by a factor of about 6.6[1] (roughly 3 LSBs). This means that we can expect about 13.5 stable bits at a gain of 128 and at a notch frequency of 10 Hz.

We also see from Tables I and II in the data sheet that a programmed gain of unity yields about 18.5 stable bits at an output rate of 10 Hz. Using the same configuration as in Figure 9, we can implement an external amplifier using two AD797 operational amplifiers. Setting the external gain to 128 yields an RTI LSB size of 74 nV for 19-bit resolution (5 V/128/2^{19}). With a 0.1 Hz to 10 Hz RTI noise of 50 nV (p-p), the two operational amplifiers will contribute a combined RTI noise of about 70 nV (p-p) that gets reduced to 28.8 nV by the 2.62 Hz cutoff of the digital filter. Because this value is less than half the LSB size of 74 nV, the presence of the external amplifier will not introduce significant noise into the system.

These calculations are based on internal device noise only. In practice power supply noise and/or noise from the sensor may make it impossible to achieve this resolution.

CHOOSING A SUITABLE REFERENCE

While errors due to the initial accuracy and drift of a voltage reference can be compensated using calibration, it is critical that the output voltage noise of the reference does not significantly degrade performance.

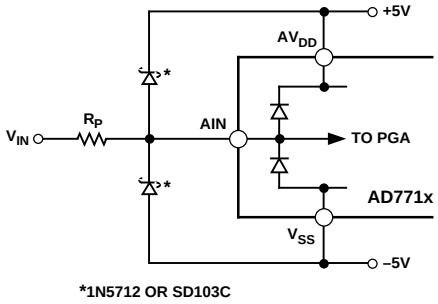
Table I in the data sheet lists the rms output noise of the converter as a function of gain and first notch frequency. The rms noise of the reference used (bandlimited to the –3 dB bandwidth of the digital filter) should be quadratically summed with this value to yield the overall noise of the system. Actual measurements in systems where reference noise is of the same order of magnitude as AD771x noise have shown, however, that the reference contributes less noise than expected.

As an example of this effect, which is not fully understood, consider the effect of the internal reference of the AD7710 on the overall noise at a gain of 1 and at a first notch frequency of 10 Hz (3 dB cutoff point is at 2.62 Hz). The internal reference has a p-p noise

of 50 μV (8.3 μV rms[1]) in the 0.1 Hz to 10 Hz bandwidth. This corresponds to a noise spectral density of $2\mu V / \sqrt{Hz}$ ($8.3\mu V / \sqrt{10Hz \times 159}$). In the 2.62 Hz bandwidth of the ADC, the reference will therefore contribute 3.4 μV rms of noise to the internal ADC noise of 1.7 μV. Quadratically summing these two values yields a total noise of 3.8 μV rms which should reduce the effective resolution from 21.5 bits to 20.5 bits. In practice, however, it was observed that the effective resolution only degraded to about 21 bits when the internal reference was used.

INPUT PROTECTION

Because exposure to overvoltage can permanently damage these devices, it is important to protect the inputs in applications where the absolute input voltage (normal-mode or common-mode) can exceed the Absolute Maximum Ratings (AV_{SS} – 0.3 V to AV_{DD} + 0.3 V) even if the overvoltage condition is momentary. The input stage of the ADC does however offer a feature that can be used as part of an input protection scheme.



$$R_P = \frac{V_{IN\,(max)} - (AV_{DD} + 0.7\,V)}{5\,mA}$$

Figure 10. Input Overvoltage Protection

As shown in Figure 10 the analog inputs are connected through ESD protection diodes to both power supply rails. By adding a current limiting resistor (R_P), the analog input can be clamped to within 0.6 V of either supply. The protection resistor should be chosen such that the input current during overvoltage does not exceed about 5 mA. If the Johnson noise due to R_P is excessive, its value may be reduced by using external Schottky diodes which clamp at 0.3 V above the supplies and which can safely conduct larger currents.

INTERFACING TO NOISY LOGIC

In addition to causing possible long-term damage to the device, overshoot or undershoot in the digital signals driving the converter, can cause communications errors. For example, an undershoot of more than 0.3 volts below ground on the SCLK line is sufficient to turn on parasitic diodes between the device and the substrate. This disturbance, in addition to adversely affecting the integrity of the conversion, can cause the ADC interface to detect multiple clock edges for each clock edge applied. This will have the effect of clocking incorrect data into the device.

This overshoot can be reduced by placing a small resistor in series on the digital line that is causing the problem. This resistance will combine with the parasitic capacitance of the digital input to form a low-pass filter that should eliminate any ringing on the signal. Typically a resistor size of 50 Ω is recommended although some experimentation may be necessary. It may also be necessary to add an external capacitance from the input to ground if the parasitic capacitance of the digital input is not large enough. Here again experimentation is necessary but a good starting point would be around 10 pF.

USING OPTO-ISOLATORS

When an isolated digital interface is required, opto-isolators can be used to create a simple and cheap isolation barrier. As shown in Figure 11, this can be implemented with as few as three opto-isolators when the AD7714 is used.

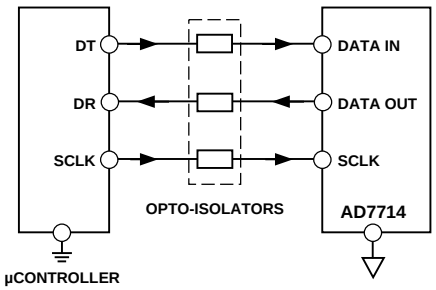


Figure 11. Isolated 3-Wire Interface

However, using opto-isolators which have relatively slow rise and fall can cause problems even when the serial clock is running at a slow speed.

CMOS logic inputs such as the SCLK and DATA IN inputs on the AD771x series are designed to be driven by either a logic zero or a logic one. In these states they source/sink a minimum amount of current. However while the input voltage is in the dead-band between logic zero and logic one (0.8 V to 2.0 V), the logic input will sink an increased amount of current. If opto-isolators are used that have relatively slow rise and fall times, the excessive amount of time spent in the dead-band will cause self-heating in the ADC. This self-heating tends to shift the threshold voltage of the logic gate which can lead to a single clock edge being interpreted by the ADC as multiple clock pulses. To prevent this threshold jitter, the SCLK lines coming from the opto-isolators should be buffered using a Schmitt trigger.

Table I. $\overline{\text{DRDY}}$ Status

Event	Output Status
Power On	$\overline{\text{DRDY}}$ goes High after Power-ON, goes Low after three Notch Periods and stays low until data are read.[1]
Wake-Up After Power Down	$\overline{\text{DRDY}}$ goes High after Wake-Up, goes Low after three Notch Periods and stays low until data are read.[1]
Voltage Step at Input	Valid data after three to four $\overline{\text{DRDY}}$s.[2]
Switch Channel	Resets $\overline{\text{DRDY}}$[3] (high). Next $\overline{\text{DRDY}}$ after three Notch Periods.
Self-Calibration	Resets $\overline{\text{DRDY}}$[3] (high). Next $\overline{\text{DRDY}}$ after nine Notch Periods.
System Calibration	Resets $\overline{\text{DRDY}}$[3] (high). Next $\overline{\text{DRDY}}$ after six Notch Periods.
System Offset Calibration	Resets $\overline{\text{DRDY}}$[3] (high). Next $\overline{\text{DRDY}}$ after nine Notch Periods.
Background Calibration	Resets $\overline{\text{DRDY}}$[3] (high). First $\overline{\text{DRDY}}$ after six Notch Periods. Valid Data after second $\overline{\text{DRDY}}$[4]. Output data rate is reduced by 1/6.
Write New Cal. Coefficients	Valid data after three to four $\overline{\text{DRDY}}$s.[2]
Change Notch Frequency	Valid data after three to four $\overline{\text{DRDY}}$s.[2]
Turn On/Off B/O Current	Valid data after three to four $\overline{\text{DRDY}}$s.[2]
Turn On/Off RTD Current[5]	Valid data after three to four $\overline{\text{DRDY}}$s.[2]
Turn On/Off CJC Current[6]	Valid data after three to four $\overline{\text{DRDY}}$s.[2]

NOTES
[1] The $\overline{\text{DRDY}}$ pin on the AD7714 returns high just before the Output Register is updated even if the old data have not been read.
[2] Asserting $\overline{\text{SYNC}}$ during or directly after the operation/event reduces the delay by up to one notch period.
[3] $\overline{\text{DRDY}}$ will not be reset high if $\overline{\text{TFS}}$ and $\overline{\text{SYNC}}$ are tied together.
[4] In Background Calibration mode, a measurement of the input voltage is interlaced alternately with either a zero-scale or a full-scale calibration, hence the need to wait for two $\overline{\text{DRDY}}$s after the calibration command has been issued before reading the first valid data.
[5] AD7711 and AD7713 only.
[6] AD7710 only.

USING $\overline{\text{DRDY}}$ AS A STATUS INDICATOR

The $\overline{\text{DRDY}}$ logic output indicates that a new conversion result is available in the output register. However as shown in Table I, this conversion result is not always valid. When a step change occurs at the input for example (this could be caused by an actual step in the input voltage or by the turning on or off of one of the onboard current sources such as the Burn-Out Current), the next three conversion results (indicated by a falling edge on $\overline{\text{DRDY}}$) are invalid due to the internal pipeline delay of the converter's digital filter. This can pose problems in systems whose interrupt lines are driven by $\overline{\text{DRDY}}$. One possible solution is to pulse the $\overline{\text{SYNC}}$ input directly after the event (i.e., turning on the Burn-Out current) has occurred. This has the effect of setting all nodes of the digital filter to zero and of keeping the $\overline{\text{DRDY}}$ line high until the output register contains a valid result. Pulsing the $\overline{\text{SYNC}}$ input also reduces the delay, to valid data, to exactly 3 notch periods.

When the device receives a calibration command, the $\overline{\text{DRDY}}$ output is driven high within one modulator cycle (1/19.5 kHz) and stays high until the calibration sequence is complete and the device has a valid conversion result in its output register. The duration of the calibration varies from 6 to 9 times the output rate depending on which calibration mode is used. Because it is necessary to recalibrate after changing the output rate (i.e., the notch frequency), it is *not* possible to save time by calibrating at a high output rate and then operating at a lower output rate. It should be noted that if the $\overline{\text{SYNC}}$ and $\overline{\text{TFS}}$ lines are tied together, this has the disadvantage that $\overline{\text{DRDY}}$ is not driven high when the calibration command is used.

USING CALIBRATION TO COMPENSATE FOR TEMPERATURE DRIFT AND REFERENCE INACCURACY

In microcontroller or microprocessor based systems where EPROM data storage is available, it is possible to calibrate the ADC to a very high accuracy even when the local reference voltage is inaccurate and drifts over temperature. In Figure 12, a reference voltage of 2.5 V is generated by driving one of the RTD current sources of the AD7711 through a 12.5 kΩ resistor. This current, which has an initial but repeatable inaccuracy of ±20% and a temperature drift of 20 ppm/°C, is also used to excite a thermistor (located close to or on the AD7711) which measures the ambient temperature. Apart from the temperature coefficient of the 12.5 kΩ resistor, the voltage generated across the thermistor (which is connected to AIN2), will be drift-free relative to the reference voltage.

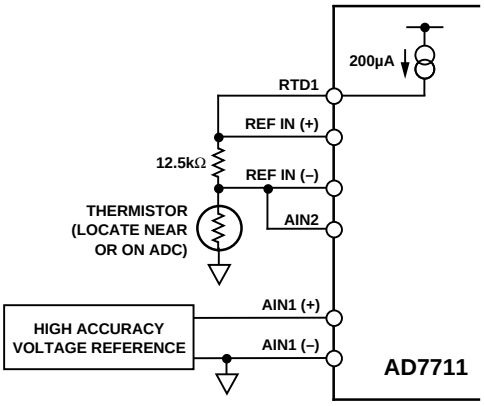


Figure 12. High Accuracy Calibration Using AD7711

Figure 13 shows the steps involved in calibrating the device over a particular temperature range. As an example let's assume that we want to calibrate over the temperature range 0°C to +70°C. After power-up, the ambient temperature is set to 5°C. A system calibration is performed on the AIN1 channel using a drift free, high accuracy voltage reference. The zero-scale (ZS) and full-scale (FS) calibration coefficients generated are read back and stored. Next we perform a self-calibration on AIN2 and then measure and store the subsequent conversion result from this channel.

The temperature is increased and the same procedure is repeated at 15°C, 25°C and so on up to 65°C. We end up with a look-up table containing seven sets of calibration coefficients corresponding to seven temperature ranges, each range covering 10°C.

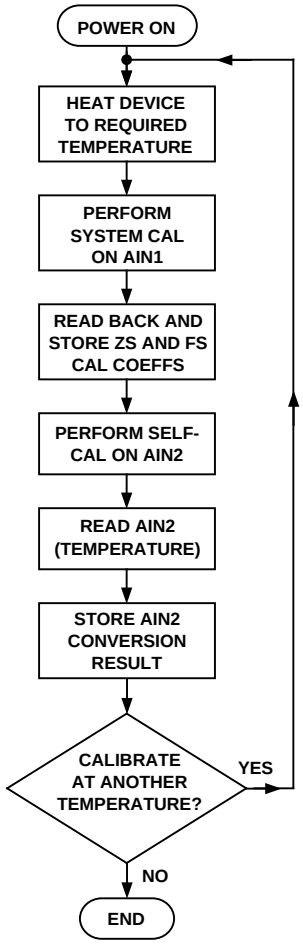


Figure 13. Flowchart for Calibration Over Temperature

Figure 14 details how to use this look-up table to perform precision measurements even while the ambient temperature is varying by a large amount. After power-up, AIN2 is self-calibrated and measured. By comparing the measured result with the values in the look-up table, we can determine in which temperature range we are operating. Finally we load the appropriate calibration coefficients from the look-up table and measure the voltage on AIN1. It is important to note that no calibration is performed on AIN1 during normal operation.

If we periodically monitor the ambient temperature by measuring AIN2 (it is probably not necessary to measure AIN2 as often as AIN1), we can load a different set of calibration coefficients if the temperature drifts outside the band in which it is currently operating.

The above example calibrated the device in 10°C segments (i.e., Calibration Temperature ±5°C). Drift occurring within these segments will not be corrected. If higher accuracy is required, smaller segments can be selected.

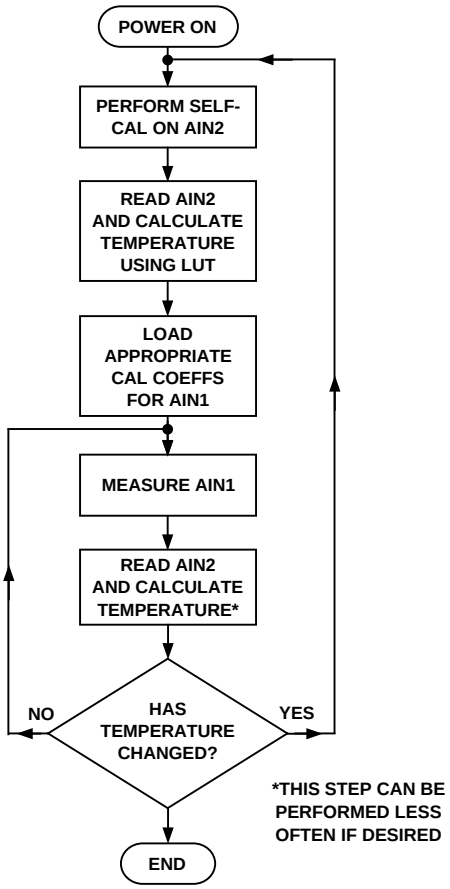


Figure 14. Flowchart for Low Drift Measurement Over Temperature Using a Look-Up Table

Figure 15 shows a flowchart for an alternative calibration scheme that relies more on self-calibration which is less labor-intensive than system calibration. Initially the system is powered on, and heated up to a temperature at which the external reference voltage is most accurate (probably room temperature). At this temperature, both a self-calibration and a system calibration are performed, the calibration coefficients being read back in each case. Next, the ratios of the self-cal coefficients to the system-cal coefficients are calculated. For more information on the structure of the calibration coefficients and on calculating this ratio, contact Analog Devices Applications.

The system is now heated up to the temperatures at which calibration is required. A self-calibration is performed at each temperature. Using the ratios previously calculated, it is now possible to use the self-cal coefficients to calculate the effective system-calibration coefficients which are based on a calibration voltage from a highly accurate voltage reference (not from the relatively inaccurate combination of a 200 μA current source and a 12.5 kΩ resistor). At this point the self-calibration coefficients can be discarded. As before, it is necessary to self-calibrate and measure the voltage on AIN2 (ambient) at each temperature. These values are stored in a look-up table along with the calculated system calibration coefficients. For operational conditions, the same scheme as described in Figure 14 is used.

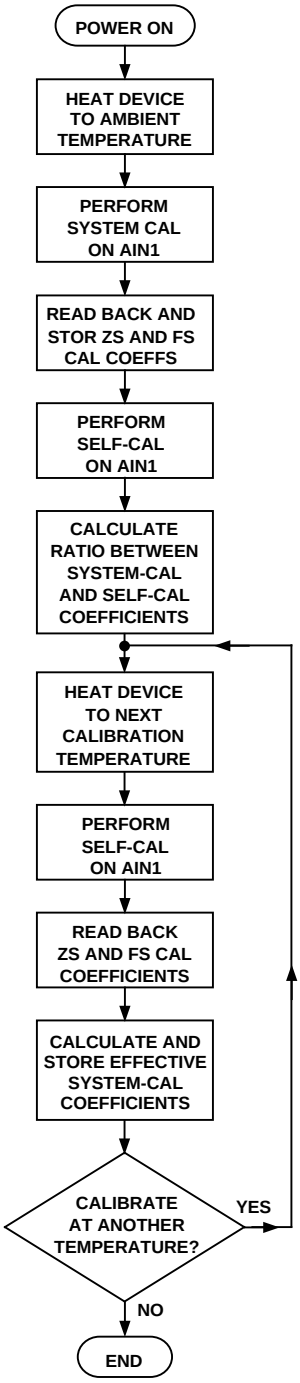


Figure 15. Using Self-Calibration as an Alternative to System Calibration

DEFAULT REGISTER SETTINGS ON POWER-UP

Table II lists the default settings of the control registers of the AD7710/AD7711/AD7712/AD7713 and of the filter register of the AD7714. These values can be read back on power-up as a useful test of both the functionality of the ADC and of the communication with the processor to which it interfaces.

Table II. Register Settings after Power-On or Reset

Device	Control Register
AD7710	000146 Hex
AD7711	000146 Hex
AD7712	000146 Hex
AD7713	000041 Hex

Device	Filter Register
AD7714	0140 Hex

References

[1] *Analog-Digital Conversion Handbook*, 3d. edition, pp. 46–47, 1986, by the Engineering Staff of Analog Devices, Inc., edited by Daniel H. Sheingold, Prentice-Hall, Englewood Cliffs, NJ 07632

APPENDIX A

```
*This program contains subroutines to read and write
*to the AD7710 family of ADCs from the 68HC11
*microcontroller. These subroutines were developed for the
*68HC11 Evaluation Board, which is where the references to
*BUFFALO come from, in conjunction with the AD7710 Evaluation
*board. The following connections need to be made.
*68HC11      AD7710
*PC0         RFS
*PC1         TFS
*PC2         DRDY
*PC3         A0
*PD2,PD3     SDATA        10K pull-up resistor
*                         PD2 and PD3 attached together
*PD4         SCLK         10K pull-up
*PD5                      10K pull-up, no connection to AD7710
*
portc  equ   $1003
portd  equ   $1008
ddrd   equ   $1009
spcr   equ   $1028
spsr   equ   $1029
spdr   equ   $102a
ddrc   equ   $1007
*
       org   $C000
read   lds   #$CFFF       sub-routine to read from the AD771x
*
       ldaa  #$fb   initialize port c outputs:  11111011
       staa  ddrc   Set up drdy as input (PC2) and
*                   A0, RFS, TFS (PC3,PC0,PC1) as outputs
       ldaa  #$30    00110000
       staa  ddrd    MOSI is low for input,
*                    MISO is high, SCK as output
*
       ldaa  #$37    00110111
       staa  spcr    SPI system off, resets itself
*
       ldaa  #$77    01110111
       staa  spcr    Interrupts disabled, SPI system on,
*                    DWOM mode, 68hc11 is master,
*                    CPOL 0, CPHA 1, SCK=ECK/32
*
       ldy   #$1000
       bset  portc,y  $03 TFS and RFS set high
*
       bset  portc,y  $08 A0 high to read data
*      bclr  portc,y  $08 A0 low to read from cal. reg
*
       ldaa  spsr  Initial dummy read to clear port and SPIF
       ldaa  spdr
*
       ldab  #$03   b is 0 when write finished
       ldx   #$00   x points to start of 24 bit word
*                    to be read
*
pause  ldaa  #$04
       anda  portc
       bne   pause  Wait until DRDY is low
*
       bclr  portc,y  $01  Clear RFS
*
```

```
go1    staa  spdr    Start SCK
wait1  ldaa  spsr
       bpl   wait1   wait until SPIF flag is clear
       ldaa  spdr    and then read.
       staa  0,x     And then put in memory
*
       decb
       beq   fin1
       inx          points to next byte to be read
       jmp   go1
*
fin1   bset  portc,y  $09  set RFS and A0
*
       jmp   $e000  Return to BUFFALO

*
*
*
write  lds   #$cfff
       ldaa  #$fb
       staa  ddrc     Set up drdy as input (PC2) and
*                     A0, RFS, TFS (PC3,PC0,PC1) as outputs
       ldaa  #$37
       staa  spcr     SPI system off, resets itself
*
       ldaa  #$73
       staa  spcr     Interrupts disabled, SPI system on,
*                     DWOM mode, 68hc11 is master,
*                     CPOL=0, CPHA=0, SCK=ECK/32
       ldaa  #$38
       staa  ddrd     MOSI is high for output,
*                     MISO is low, SCK is high
       ldy   #$1000
       bset  portc,y  $03   Set TFS and RFS
*
       bclr  portc,y  $08   Set A0 low to write
*                            to control register
*
*       ldaa  spsr     Initial dummy read to clear
*       ldaa  spdr     port and SPIF
*
*
       ldab  #$03     b is 0 when write finished
       ldx   #$00     x points to start of 24 bit word
*                     to be written
*
       bclr  portc,y  $02  clear TFS
*
go2    ldaa  0,x
       staa  spdr     write byte to serial port
*
wait2  ldaa  spsr
       bpl   wait2    wait until SPIF flag is clear
*
       decb
       beq   fin2
       inx            points to next byte to be written
       jmp   go2
*
fin2   bset  portc,y  $0a    set TFS and A0
*
       jmp  $e000     Return to BUFFALO
```

E2113–10–6/96

PRINTED IN U.S.A.