Analog-to-Digital Converters

85.1 Introduction
85.2 Sampling
85.3 Quantization
85.4 ADC Specifications
Range and Resolution • Coding Conventions • Linear Errors • Nonlinear Errors • Aperture Errors • Noise • Dynamic Range
85.5 Types of ADCs
Flash • Successive-Approximation Register • Multistage • Integrating • Sigma–Delta ADCs • Voltage-to-Frequency Converters
85.6 Instrumentation and Components
Integrated Circuits • Instrumentation

E. B. Loewenstein
National Instruments

85.1 Introduction

Almost every modern instrumentation system includes some form of digitizer, or *analog-to-digital converter (ADC)*. An ADC converts real-world signals (usually voltages) into digital numbers so that a computer or digital processor can (1) acquire signals automatically, (2) store and retrieve information about the signals, (3) process and analyze the information, and (4) display measurement results. A digitizing system can do these jobs with greater speed, reliability, repeatability, accuracy, and resolution than a purely analog system normally can.

The two main functions of an ADC are *sampling* and *quantization*. These two processes convert analog signals from the time and voltage continuums (respectively) into digital numbers having discrete amplitudes, at discrete times. To represent changing signals at every instant in time or at every possible voltage would take an infinite amount of storage. So for every system there is an appropriate *sampling rate* and degree of quantization (*resolution*) so that the system retains as much information as it needs about the input signals while keeping track of manageable amounts of data. Ultimately, the purpose of sampling and quantization is to reduce as much as possible the amount of information about a signal that a system must store in order to reconstruct or analyze it meaningfully.

85.2 Sampling

To prevent having to digitize an infinite amount of information, an analog signal must first be sampled. Sampling is the process of picking one value of a signal to represent the signal for some interval of time. Normally, digitizers take samples uniformly in time, e.g., every microsecond. It is not necessary to sample uniformly, but doing so has some interesting and convenient mathematical properties, which we will see later.

Sampling is done by a circuit called a *sample-and-hold (S/H)*, which, at a sampling instant, transfers the input signal to the output and holds it steady, even though the input signal may still be changing. An S/H usually consists of a signal buffer followed by an electronic switch connected to a capacitor. At a sampling instant, the switch briefly connects the buffer to the capacitor, allowing the capacitor to charge to the input voltage. When the switch is disconnected, the capacitor retains its charge and thus keeps the sampled input voltage steady while the ADC that follows does its job. Quite often, sampling is actually done by a circuit called a *track-and-hold (T/H)*, which differs from an S/H only slightly. Whereas the S/H holds the analog signal until the next sampling instant, the T/H holds the analog signal still only until the ADC has finished its conversion cycle. After the ADC is through, the T/H reconnects the buffer to the capacitor and follows the input signal until the next sampling instant. The result is more accurate sampling, because the buffer has more time to charge the capacitor and "catch up" with (track) the input signal, which has changed since the last sampling instant. Nearly every modern ADC chip has a built-in S/H or T/H, and virtually all data acquisition systems include them.

Of course, sampling necessarily throws away some information, so the art of sampling is in choosing the right sample rate so that enough of the input signal is preserved. The major pitfall of *undersampling* (sampling too slowly) is *aliasing*, which happens whenever the input signal has energy at frequencies greater than one-half the sample rate. In Figure 85.1a, a signal (the fast sine wave) is sampled at a rate

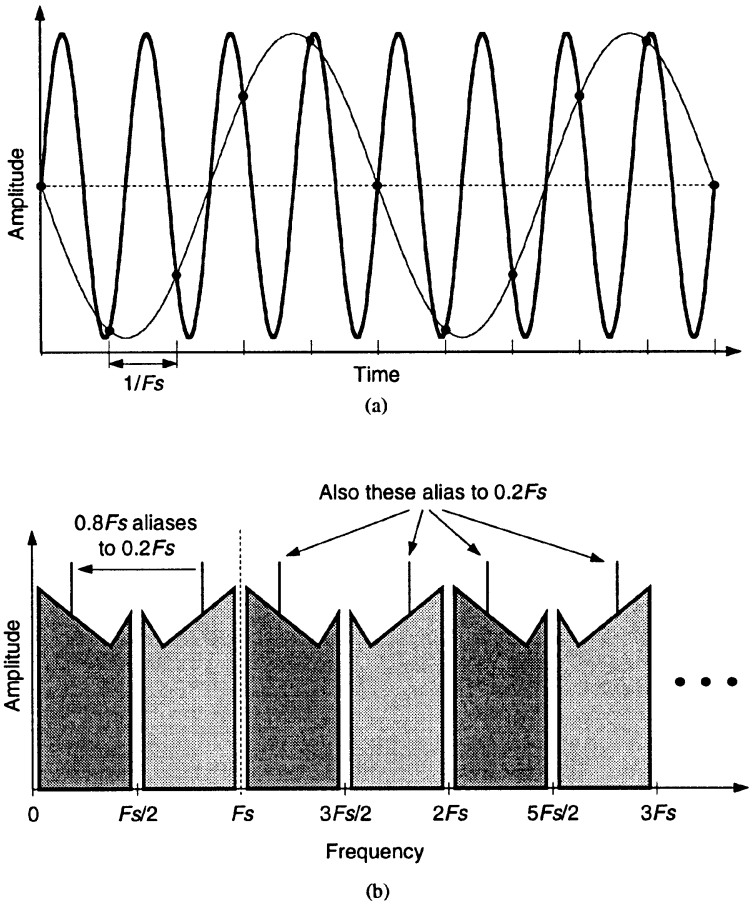


FIGURE 85.1 A demonstration of aliasing. An ADC sampling at rate *Fs* cannot distinguish between a 0.8*Fs* sine wave and a 0.2*Fs* sine wave. (a) A time-domain illustration. (b) A frequency-domain illustration. Theoretically, a sampler aliases an infinite number of 0.5*Fs*-wide frequency bands into the baseband (0 to 0.5*Fs*). Practically, finite analog bandwidth eventually limits how far out in frequency aliases can come from.

Fs, shown by the hash marks at the bottom of the graph. The sine wave has a frequency of 0.8*Fs*, which is higher than one half the sample rate (0.5*Fs*). Notice that sampling the lighter sine wave of 0.2*Fs* produces the same set of samples. The resulting sampled data is ambiguous in that we cannot tell from the data what the frequency of the incoming sine wave actually is. In fact, even though the data set appears to represent a sine wave of 0.2*Fs*, the actual signal could be any sine wave having a frequency of $(n)Fs \pm 0.2Fs$, where *n* is any integer, starting with 0. So the original signal could be 0.2*Fs*, 0.8*Fs*, 1.2*Fs*, 1.8*Fs*, 2.2*Fs*, etc. (or even more than one of those). We say that 0.2*Fs* is the *alias* of a signal that may actually be at another frequency entirely. During interpretation of sampled data, it is customary to treat signals as though they occurred in the baseband (0 to 0.5*Fs*), whether or not that is the case. In general, in a system sampling at *Fs*, a signal at a frequency *F* will alias into the baseband at

$$Fa = \text{abs}\left[\left(\text{n}\right)Fs - F\right], \tag{85.1}$$

where abs denotes absolute value, $n \geq 0$, and $(n)Fs$ is the closest integer multiple of *Fs* to *F*.

Everyone has seen a demonstration of aliasing at the movies, in the form of "wagon-wheeling." As the stagecoach or wagon takes off, the wheels begin to turn, slowly at first, then faster. As the wagon speeds up, the spokes suddenly appear to be turning backward, even though the wagon is moving forward. Sometimes the spokes appear to be standing still. The reason for this is that a motion picture camera shooting film at 24 frames/s is a sampling system operating at 24 samples/s. The turning wagon wheel is a periodic signal that the camera undersamples. When the wheel begins turning just fast enough that one spoke travels at least half the distance to the next spoke in 1/24th of a second, the spokes begin to appear to move backward, and the system is aliasing. When the wheel is turning so that a spoke moves exactly the distance between two spokes in 1/24th of a second, the spokes appear to be standing still, since they all look the same to the camera.

It follows from Equation 85.1 that if we put into a sampler a signal with no energy at frequencies greater than one half the sample rate (0.5*Fs*), then aliasing will not occur. This is the essence of the Shannon sampling theorem [1], which states that, with mathematical interpolation, the complete input waveform can be recovered *exactly* from the sampled data, at all times at and in between the sampling instants, as long as the sample rate is at least twice as high as the highest frequency content in the signal. Sometimes we refer to 0.5*Fs* as the *Nyquist frequency*, because Nyquist was concerned with the maximum bandwidth of signals [2]. Similarly, twice the highest frequency content of a signal (i.e., the minimum nonaliasing sample rate) is sometimes called the *Nyquist rate*. Sample rates are specified in samples/s, or S/s, and it is also common to specify rates in kS/s, MS/s, and even GS/s.

It is not always necessary to worry about aliasing. When an instrument is measuring slow-moving dc signals or is gathering data for statistical analysis, for instance, getting frequencies right is not important. In those cases we choose the sample rate so that we can take enough data in a reasonable amount of time. On the other hand, if the instrument is a spectrum analyzer, where frequency does matter, or an oscilloscope, where fine time detail is needed, aliasing certainly is an issue. When aliased signals from beyond the frequency band of interest can interfere with measurement, an instrument needs to have an *antialias filter* before the S/H. An antialias filter is a low-pass filter with a gain of 1 throughout most of the frequency band of interest. As frequency increases, it begins to attenuate the signal; by the Nyquist frequency it must have enough attenuation to prevent higher-frequency signals from reaching the S/H with enough amplitude to disturb measurements. An efficient antialias filter must attenuate rapidly with frequency in order to make most of the baseband usable. Popular analog filters with rapid cutoff include elliptic and Chebyshev filters, which use zeros to achieve fast cutoff, and Butterworth filters (sixth order and above), which do not attenuate as aggressively, but have very flat passband response. A good book about filters is Reference 3.

Some ADCs do not need a S/H or T/H at all. If the ADC is converting a slow-moving or dc signal and precise timing isn't needed, the input may be stable enough during conversion that it is as good as

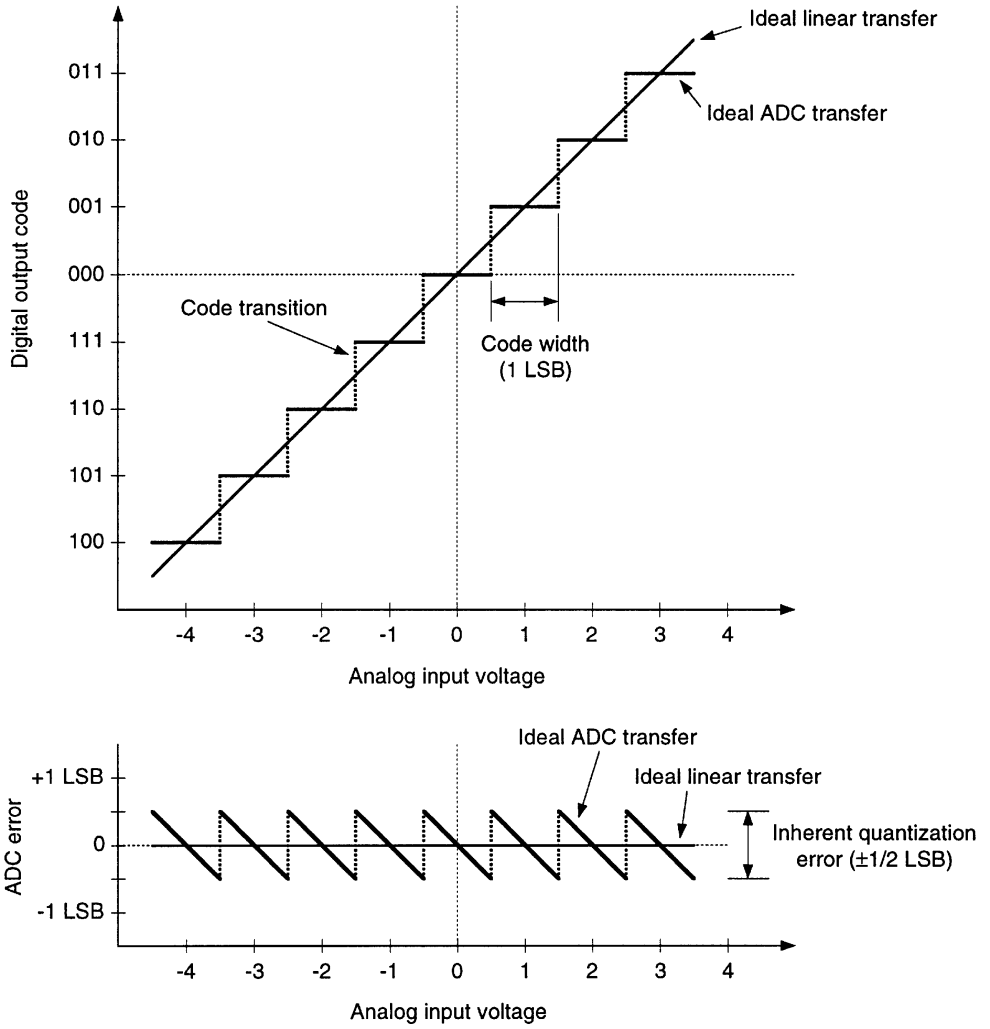


FIGURE 85.2 The ideal three-bit quantizer has eight possible digital outputs. The analog input-to-digital output transfer function is a uniform staircase with steps whose width and height are 1 LSB exactly. The bottom graph shows the ideal transfer function (a straight line) subtracted from the staircase transfer function.

sampled. There are also *integrating ADCs* (discussed later), which average the input signal over a period of time rather than sampling it. However, internally they actually sample the average.

85.3 Quantization

What sampling accomplishes in the time domain, quantization does in the amplitude domain. The process of digitization is not complete until the sampled signal, which is still in analog form, is reduced to digital information. An ADC quantizes a sampled signal by picking one integer value from a predetermined, finite list of integer values to represent each analog sample. Each integer value in the list represents a fraction of the total analog input range. Normally, an ADC chooses the value closest to the actual sample from a list of uniformly spaced values. This rule gives the *transfer function* of analog input-to-digital output a uniform "staircase" characteristic. Figure 85.2 represents a three-bit quantizer, which maps a continuum of analog input values to only eight (2^3) possible output values. Each step in the staircase has (ideally) the same width along the *x*-axis, which we call *code width* and define as 1 *LSB (least*

significant bit). In this case 1 LSB is equal to 1 V. Each digital code corresponds to one of eight 1-LSB intervals making up the analog input range, which is 8 LSB (and also 8 V in this case).

Of course, we would like our measurement system to have a transfer function that is a straight line and has no steps at all. The bottom graph in Figure 85.2 is the ideal transfer function (a straight diagonal line) subtracted from the staircase function, or the *quantization error*. In an ideal ADC, the quantization error is bounded by ±½ LSB, and, over the input range, the average error is 0 LSB and the standard deviation of error is $1/\sqrt{12}$ LSB. As the bottom graph shows, the quantization error at any point is a deterministic function of the input signal.

85.4 ADC Specifications

Range and Resolution

The *input range* of an ADC is the span of voltages over which a conversion is valid. The end points at the bottom and the top of the range are called *–full-scale* and *+full-scale*, respectively. When –full-scale is 0 V the range is called *unipolar*, and when –full-scale is a negative voltage of the same magnitude as +full-scale the range is said to be *bipolar*. When the input voltage exceeds the input range, the conversion data are certain to be wrong, and most ADCs report the code at the end point of the range closest to the input voltage. This condition is called an *overrange*.

The *resolution* of an ADC is the smallest change in voltage the ADC can detect, which is inherently 1 LSB. It is customary to refer to the resolution of an ADC by the number of binary bits or decimal digits it produces; for example, "12 bits" means that the ADC can resolve one part in 2^{12} (= 4096). In the case of a digital voltmeter that reads decimal digits, we refer to the number of digits that it resolves. A "6-digit" voltmeter on a 1 V scale measures from –0.999999 V to +0.999999 V in 0.000001 V steps; it resolves one part in 2 000 000. It is also common to refer to a voltmeter that measures from –1.999999 to +1.999999 as a "6½ digit" voltmeter. Figure 85.3 compares the resolutions of common word lengths for ADCs.

Coding Conventions

There are several different formats for ADC output data. An ADC using *binary* coding produces all 0s (e.g., 000 for the three-bit converter) at –full-scale and all 1s (e.g., 111) at +full-scale. If the range is bipolar, so that –full-scale is a negative voltage, binary coding is sometimes called *offset binary*, since the code 0 does not refer to 0 V. To make digital 0 correspond to 0 V, bipolar ADCs use *two's complement* coding, which is identical to offset binary coding except that the *most significant bit* (*MSB*) is inverted, so that 100 … 00 corresponds to –full-scale, 000 … 00 corresponds to 0 V (*midscale*), and 011 … 11 corresponds to +full-scale. All of the figures in this chapter depicting three-bit ADC transfer functions use two's complement coding.

Decimal-digit ADCs, such as those used in digital voltmeters, use a coding scheme call *binary-coded decimal (BCD)*. BCD data consists of a string of four-bit groups of binary digits. Each four-bit group represents a decimal digit, where 0000 is 0, 0001 is 1, and so on, up to 1001 for 9. The other six combinations (1010 through 1111) are invalid, or can be used for special information, such as the sign of the conversion.

Linear Errors

Linear errors are the largest and most common errors in an ADC and are easily corrected by simple calibrations or by additions with and multiplications by correction constants. Linear errors do not distort the transfer function; they only change somewhat the input range over which the ADC operates.

Figure 85.4 shows the transfer function of an ideal three-bit ADC with some *offset error*. The straight line joining the centers of the code transitions is raised, or offset, by 0.6 LSB, and the bottom graph shows the resulting error. Figure 85.5 shows an ideal three-bit ADC with a +25% *gain error*. The slope of the

Bits	Digits	Voltmeter "Digits"	Steps in FSR	Step size, ppm	Theoretical Dynamic Range (dB)
30	8.730		1 073 741 824	0.001	182.379
28.575	8.301	8 1/2	400 000 000	0.003	173.802
28	8.128		268 435 456	0.004	170.338
27.575	8	8	200 000 000	0.005	167.782
26	7.526		67 108 864	0.015	158.297
25.253	7.301	7 1/2	40 000 000	0.025	153.802
24.253	7	7	20 000 000	0.05	147.782
• 24	6.924		16 777 216	0.060	146.255
22	6.322		4 194 304	0.238	134.214
21.932	6.301	6 1/2	4 000 000	0.25	133.802
20.932	6	6	2 000 000	0.5	127.782
• 20	5.720		1 048 576	0.954	122.173
18.610	5.301	5 1/2	400 000	2.5	113.802
18	5.118		262 144	3.815	110.132
17.610	5	5	200 000	5	107.782
• 16	4.515		65 536	15.259	98.091
15.288	4.301	4 1/2	40 000	25	93.802
14.288	4	4	20 000	50	87.782
14	3.913		16 384	61.035	86.049
• 12	3.311		4 096	244.141	74.008
11.966	3.301	3 1/2	4 000	250	73.802
10.966	3	3	2 000	500	67.782
10	2.709		1024	976.563	61.967
8.644	2.301	2 1/2	400	2500	53.802
• 8	2.107		256	3906.25	49.926
7.644	2	2	200	5000	47.782
6	1.505		64	15625	37.885

FIGURE 85.3 Comparison of theoretical resolutions of ADCs. "Bits" refers to binary word length, and "digits" refers to decimal wordlength. • denotes popular binary word lengths. FSR is full-scale range, and theoretical dynamic range is computed from the formula 1.7609 + 6.0206*n*, where *n* is the number of bits (see discussion of dynamic range).

line through the code transitions is 1.25 times the ideal slope of 1.00. If the slope of the line were 0.75 instead, the gain error would be –25%. The bottom graph shows the error resulting from excessive gain. Offset errors can be compensated for simply by adding a correcting voltage in the analog circuitry or by adding a constant to the digital data. Gain errors can be corrected by analog circuitry like potentiometers or voltage-controlled amplifiers or by multiplying the digital data by a correction constant.

Nonlinear Errors

Nonlinear errors are much harder to compensate for in either the digital or analog domain, and are best minimized by choosing well-designed, well-specified ADCs. Nonlinearities are characterized in two ways: differential nonlinearity and integral nonlinearity.

Differential nonlinearity (DNL) measures the irregularity in the code step widths by comparing their widths to the ideal value of 1 LSB. Figure 85.6 illustrates the three-bit ADC with some irregular code widths. Most of the codes have the proper width of 1 LSB and thus contribute no DNL, but one narrow code has a width of 0.6 LSB, producing a DNL of –0.4 LSB, and one wide code has a width of 1.8 LSB, producing a DNL of +0.8 LSB at that code. This converter would be consistent with a DNL specification of ±0.9 LSB, for example, which guarantees that all code widths are between 0.1 and 1.9 LSB.

It is possible for a code not to appear at all in the transfer function. This happens when the code has a width of 0 LSB, in which case we call it a *missing code*. Its DNL is –1 LSB. If an ADC has a single missing code, the step size at that point in the transfer function is doubled, effectively reducing the local resolution of the ADC by a factor of two. For this reason it is important for an ADC specification to declare that the ADC has *no missing codes*, guaranteeing that every code has a width greater than 0 LSB. Even if an

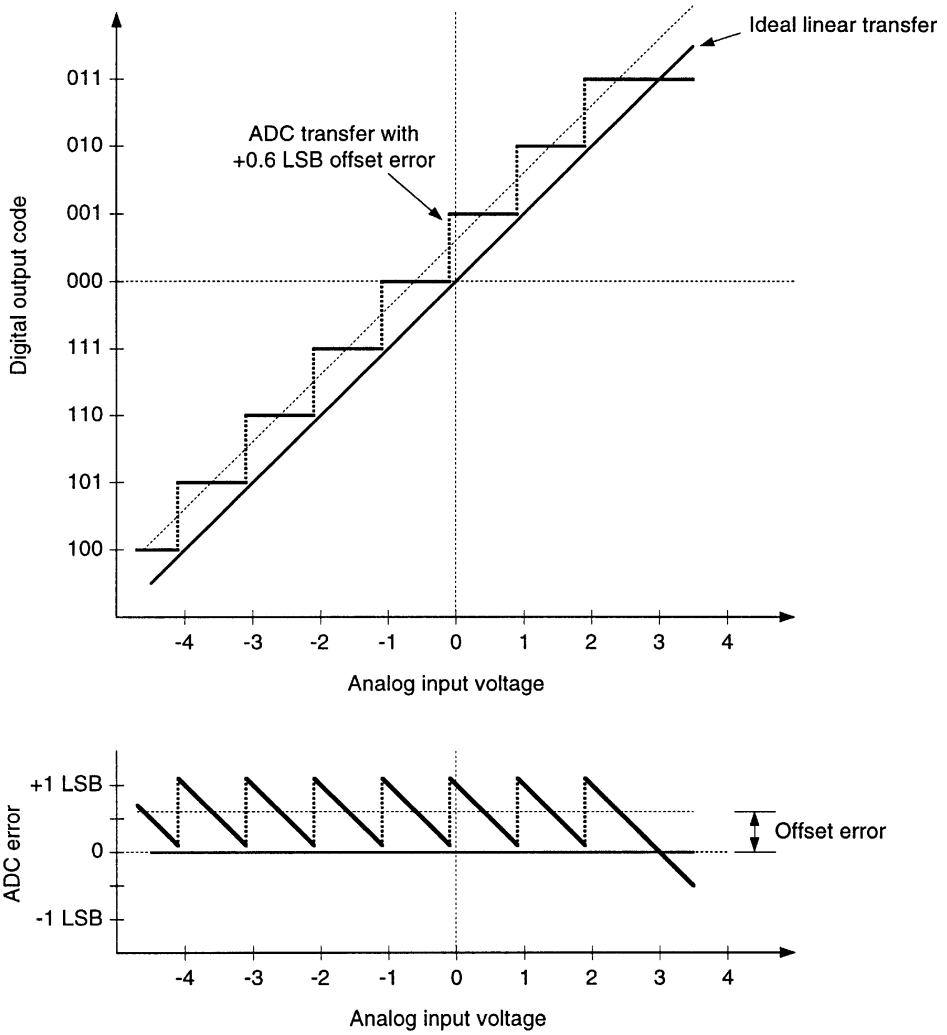


FIGURE 85.4 An ideal three-bit quantizer, only with +0.6 LSB of offset error.

ADC has missing codes, no code can have a width less than 0 LSB, so the DNL can never be worse than –1 LSB.

Integral nonlinearity (INL) measures the deviation of the code transitions from the ideal straight line, providing that the linear errors (offset and gain) have been removed. Figure 85.7 depicts an ADC with an INL error of +0.7 LSB. The offset and gain errors have been calibrated at the end points of the transfer function.

Relative accuracy (RA) is a measure of nonlinearity related to INL, but more useful. It indicates not only how far away from ideal the code transitions are, but how far any part of the transfer function, including quantization "staircase" error, deviates from ideal (assuming offset and gain errors have been calibrated at the end points). In a noiseless ADC, the worst-case RA always exceeds the worst-cast INL by ±0.5 LSB, as demonstrated in Figure 85.7. In an ADC that has a little inherent noise or has noise (called *dither*) added at the input, the RA actually improves because the addition of noise to the quantizer tends to smooth the *averaged* transfer function. Figure 85.8 shows the average of the digital output data as a function of the input voltage when 0.1 LSB rms of Gaussian random noise is intentionally added to the input. The RA improves to ±0.3 LSB from ±0.5 LSB in the noiseless case. If about 0.5 LSB rms of Gaussian noise is added, the quantization staircase becomes nearly straight. This improvement in linearity

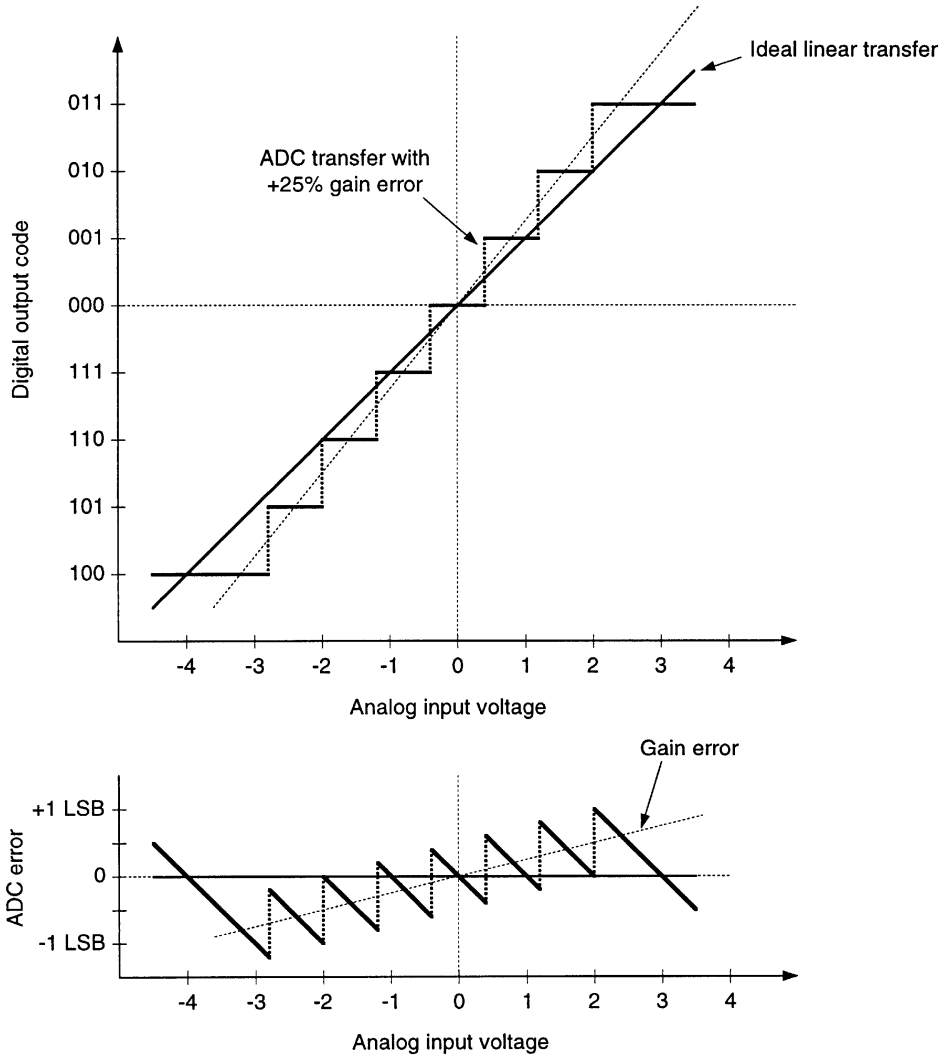


FIGURE 85.5 An ideal three-bit quantizer, only with a gain of 1.25 instead of 1.00. This represents a +25% gain error.

comes at the expense of the random error in each individual conversion caused by the noise. Adding more noise to the ADC does not improve the average quantization error much more, but it does tend to smooth out local nonlinearities in the averaged transfer function. For a good discussion of noise and dither, see Reference 4.

Aperture Errors

Aperture errors have to do with the timing of analog-to-digital conversions, particularly of the S/H. *Aperture delay* characterizes the amount of time that lapses from when an ADC (S/H) receives a convert pulse to when the sample is held as a result of the pulse. Although aperture delay (sometimes called *aperture time*) is usually specified as a few nanoseconds for an ADC or S/H by itself, this delay is usually much more than negated by the group delay in any amplifiers that precede the S/H, so that the convert pulse arrives at the S/H quite some time before the analog signal does. For instance, a typical 1 MHz bandwidth amplifier has 160 ns of delay; if the ADC or S/H it was connected to had an aperture delay of 10 ns, the effective aperture delay for the system would be –150 ns.

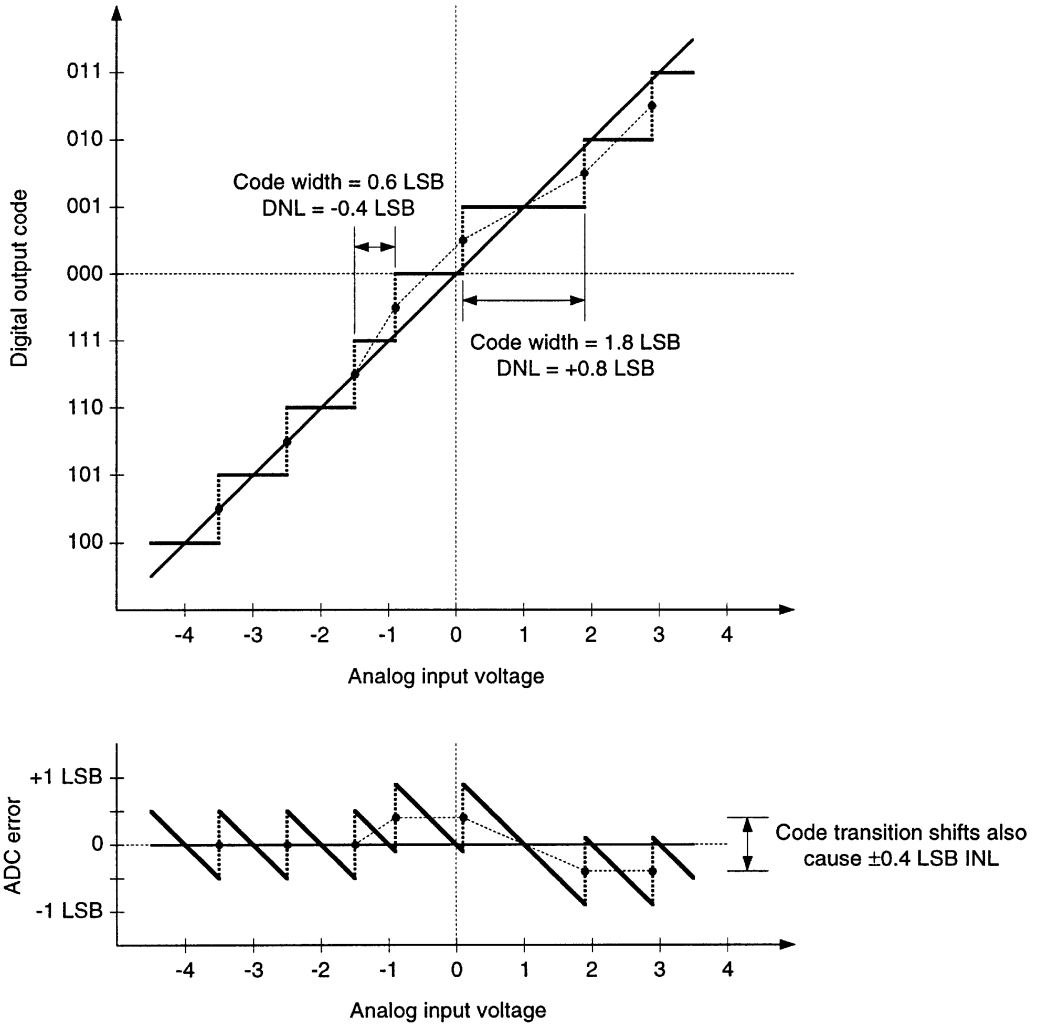


FIGURE 85.6 A three-bit quantizer with substantial DNL errors. The bottom graph illustrates the resulting INL errors.

Jitter (or *aperture jitter*) characterizes the irregularity in times at which samples are taken. If the nominal period between samples in an ADC is 1 μs, the actual time may vary from 1 μs by as much as a few hundred picoseconds or even as much as a nanosecond from cycle to cycle. Contributions to these variations can come from the crystal clock source (if included under the jitter specification), digital clock circuitry, or the S/H. Jitter is usually specified in picoseconds peak-to-peak or picoseconds rms.

Jitter interferes with measurements (particularly spectral analysis) by effectively frequency modulating the input signal by the jitter profile. A jittery ADC sampling a pure sine wave would scatter energy from the sine wave all throughout the spectrum, perhaps covering up useful spectral information. In a typical ADC, however, most of the interference from jitter tends to occur at frequencies very close to the main signal.

Noise

Noise, whether inherent in an ADC or introduced intentionally (see dither above), limits the resolution of an ADC by adding an interfering waveform to the input signal as the data is converted. Noise comes

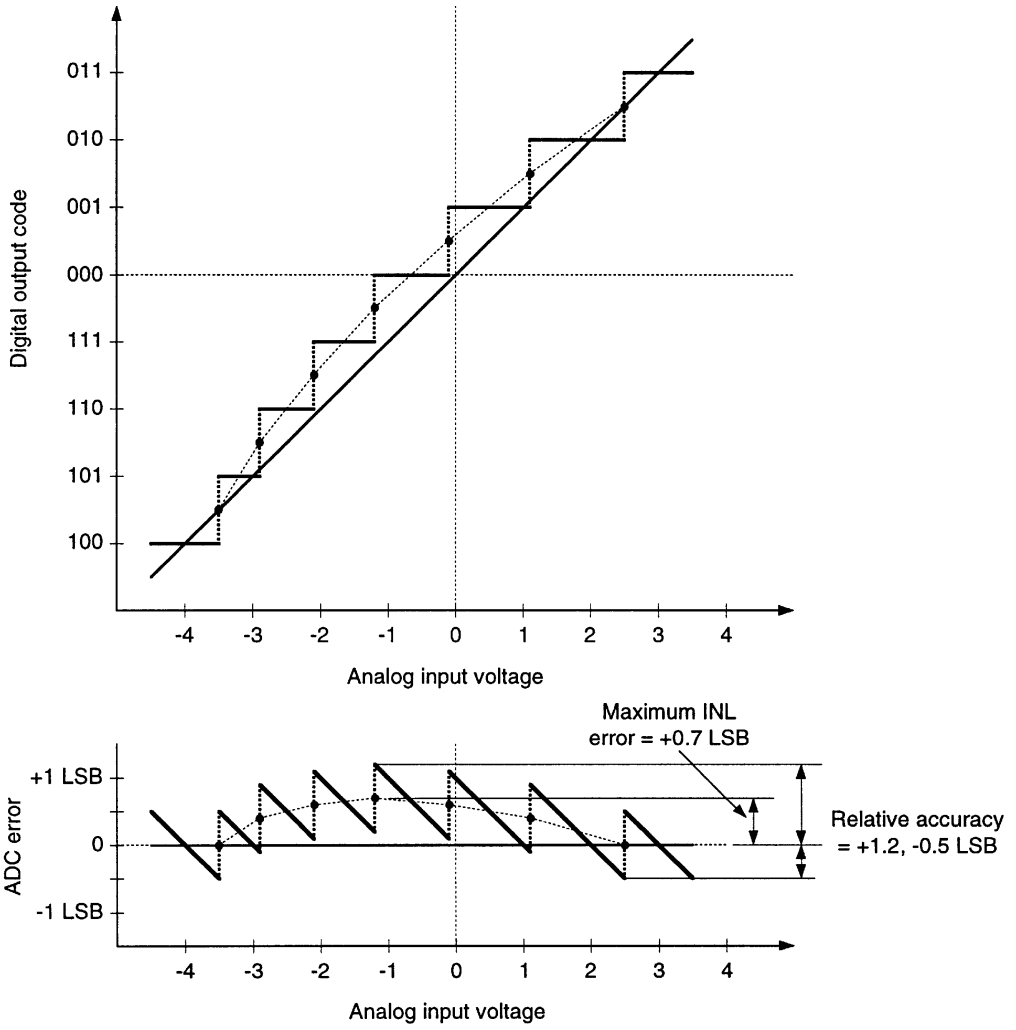


FIGURE 85.7 A three-bit quantizer with substantial INL errors. Here, the DNL error is still significant; but, for example, a 12-bit converter with 0.7 LSB of INL from a smooth error "bow" like the one above could have negligible DNL because it would have so many more steps over which to accumulate error.

from many places. The most common kind of noise is *thermal noise*, which is caused by the random nature of electric conduction in resistors and transistors. Thermal noise is worse at higher temperatures and higher resistances. Most other ADC noise is coupled electromagnetically from nearby circuitry, such as clock or logic circuits, or from routing of other input signals. Noise is usually specified in volts rms or peak-to-peak, or LSBs rms or peak-to-peak.

Quantization error (see above) can sometimes be thought of as *quantization noise*. Although quantization error is perfectly predictable with respect to the input signal, when a signal is fairly "busy" (i.e., busy enough that consecutive conversions do not tend to repeat data) the quantization error becomes chaotic, and it can be thought of it as another source of random noise, whose statistical distribution is uniform from –0.5 LSB to +0.5 LSB and whose standard deviation is $1/\sqrt{12}$ LSB. This is sometimes the dominant source of noise in spectral analysis applications.

Once noise gets into an ADC, there are ways to process out the noise if it is independent of the signal. Acquisitions of DC signals can be quieted by collecting a number of points and averaging the collection. If the noise is *white random noise*, which has equal energy density at all frequencies, averaging can reduce

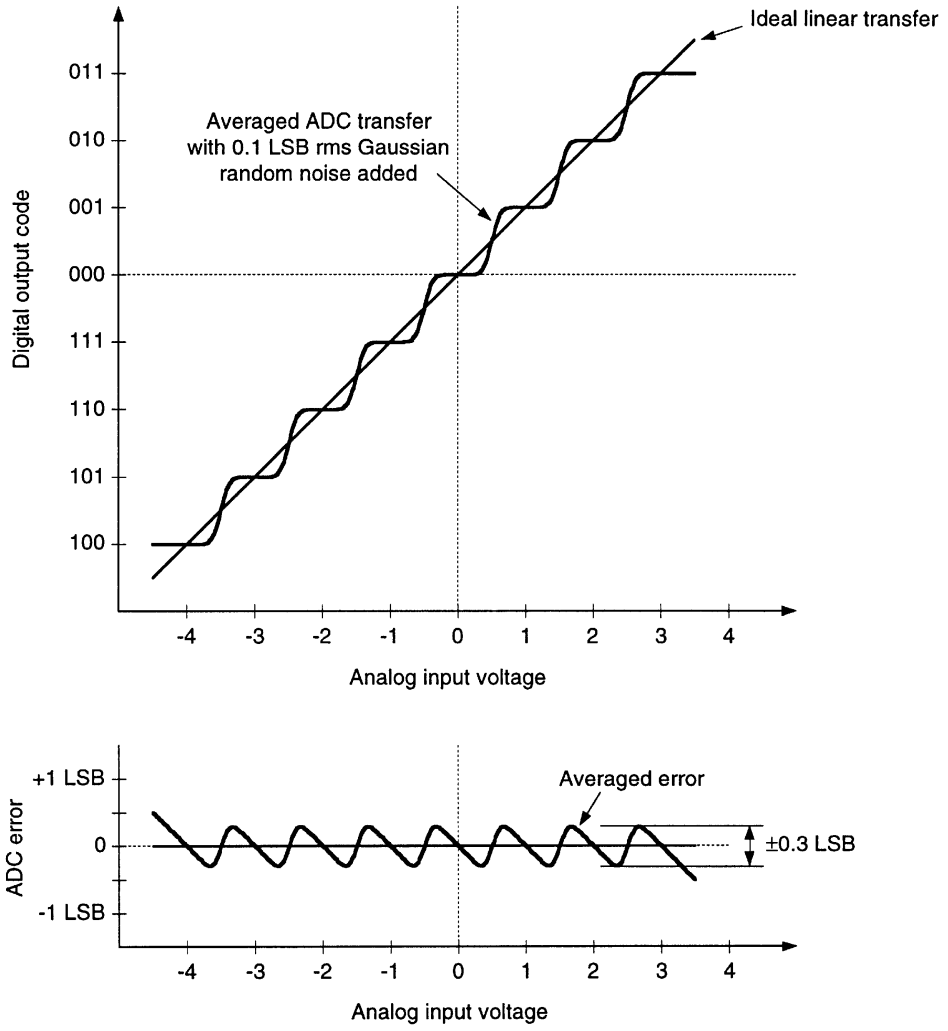


FIGURE 85.8 An ideal three-bit quantizer with 0.1 LSB rms Gaussian random noise (dither) added at the input. The relative accuracy has improved to ±0.3 LSB rms from the ±0.5 LSB expected from a noiseless quantizer. With the application of 0.5 LSB rms Gaussian noise, the transfer function becomes almost perfectly straight. Larger amounts of dither produce essentially no improvement in linearity.

the amount of noise by the square root of the number of samples averaged. The noise interfering with a repetitive waveform can be quieted by acquiring many waveforms using a level trigger and averaging the collection to produce an average waveform. Most digital oscilloscopes have waveform averaging. Quantization noise, as described above, cannot be averaged out unless other random noise is present.

The noise specifications for an ADC are for quiet, low-impedance signals at the input, such as a dead short. To preserve the noise performance of the ADCs, the user must carefully connect signals to the input with tidy cabling that keeps away from sources of electromagnetic noise. For more information on noise sources and treatment and prevention of noise, see References 5 and 6.

Dynamic Range

The *dynamic range (DR)* of an ADC is the ratio of the largest to the smallest signals the converter can represent. The largest signal is usually taken to be a full-scale sine wave, and the smallest signal is usually

taken to be the background noise level of the ADC. It can be expressed simply as a ratio, but it is more common to express it in decibels (dB):

$$DR = 20\log\left(S/N\right), \tag{85.2}$$

where DR is dynamic range in dB, S is the rms amplitude of the largest signal, and N is the rms amplitude of the smallest signal, the noise. The noise must include the quantization noise of the ADC, which for a perfect, noiseless converter is $1/\sqrt{12}$ LSB rms. For an n-bit converter, a full-scale sine wave has a peak amplitude of 2^{n-1} LSB, which corresponds to an rms amplitude of $2^{n-1}/\sqrt{2}$ LSB, or $2^{n-1.5}$ LSB rms. Hence a perfect ADC had a dynamic range of

$$\begin{aligned} DR &= 20\log\left(2^{n-1.5} * \sqrt{12}\right) \\ &= 20\log\left(2^{n}\right) + 20\log\left(2^{-1.5} * \sqrt{12}\right) \\ &= \left(n\right)\left[20\log\left(2\right)\right] + 20\log\left(\sqrt{1.5}\right) \\ &\approx 6.0206n + 1.7609. \end{aligned} \tag{85.3}$$

Equation 85.3 can be used to determine the *effective number of bits (ENOB)* of an imperfect ADC. ENOB may take only noise into account, or it may include noise and harmonic distortion products of the input signal. It is computed as

$$ENOB = \left(DR - 1.7609\right)/6.0206. \tag{85.4}$$

For example, a 16-bit ADC with a dynamic range of 92 dB has an ENOB of 14.988 bits.

85.5 Types of ADCs

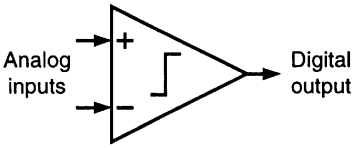
The fundamental building block of analog-to-digital conversion is the *comparator*. Every type of ADC has at least one comparator in it, and some ADCs have many. The comparator itself is a one-bit ADC; it has two analog voltage inputs and (usually) one digital output. If the voltage at the + input is greater than the voltage at the – input, the output of the comparator is a digital 1. If the voltage at the + input is less than the voltage at the – input, the output is a digital 0 (see Figure 85.9).

Another piece that all ADCs have in common is a linearity reference. This is what a comparator in an ADC compares the input signal with in the process of conversion. It directly determines the differential and integral nonlinearities of the ADC. Examples of linearity references include capacitors (in integrating ADCs) and DACs (found in successive-approximation ADCs).

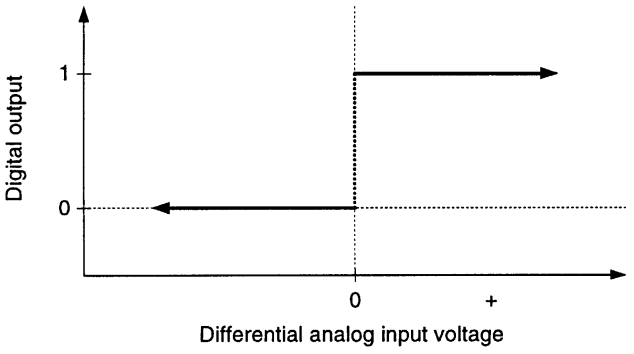
The third piece that every ADC has is a voltage reference. The reference(s) determine the full-scale input range of the ADC and are usually part of or closely associated with the linearity reference.

Flash

Flash converters are the fastest ADCs, achieving speeds near 1 GS/s and resolutions of 10 bits and below. The flash converter with n bits of resolution has $2^n - 1$ high-speed comparators operating in parallel (see Figure 85.10). A string of resistors between two voltage references supplies a set of uniformly spaced voltages that span the input range, one for each comparator. The input voltage is compared with all of these voltages simultaneously, and the comparator outputs are 1 for all voltages below the input voltage and 0 for all the voltages above the input voltage. The resulting collection of digital outputs from the



(a)



(b)

FIGURE 85.9 The comparator is the essential building block of all ADCs. (a) Comparator symbol. (b) Comparator input/output transfer function.

comparators is called a "thermometer code," because the transition between all 1s and all 0s floats up and down with the input voltage. Fast logic converts the thermometer codes to normal *n*-bit binary numbers.

Because of their simplicity, they are fast, but flash converters are limited to resolutions of 10 bits and below because the number of comparators and resistors goes up exponentially with resolution. Because the string resistor values typically vary only a few percent from each other in practical devices, the differential linearity of the flash ADC is quite good. But the same resistor variations can accumulate error across the input range and cause integral nonlinearity of a few LSB.

Successive-Approximation Register

Successive-approximation register (SAR) ADCs are the most common ADCs, having resolutions of 8 to 16 bits and speeds of 1 MS/s and below. They are generally low in cost, and they typically have very good integral linearity. The *n*-bit SAR ADC contains a high-speed *n*-bit DAC and comparator in a feedback loop (see Figure 85.11). The successive-approximation register sequences the DAC through a series of *n* "guesses," which are compared with the input voltage (Figure 85.12). As the conversion progresses, the register builds the *n*-bit binary conversion result out of the comparator outputs. By the end of the sequence the register has converged to the closest DAC value to the input voltage.

The speed of an SAR ADC is limited by the speed of the DAC inside the feedback loop. The DAC must settle *n* times to within $1/2^{-n}$ of full-scale within the conversion time of the ADC. Current SAR technology achieves 12-bit resolution at 1 MS/s and 16-bit resolution at 200 kS/s. Faster conversion at these resolutions requires multistage architectures.

Multistage

To achieve higher sample rates than SAR ADCs at resolutions of 10 to 16 bits, *multistage* ADCs (sometimes called *subranging* or *multipass* ADCs) use the iterative approach of SAR ADCs but reduce the number

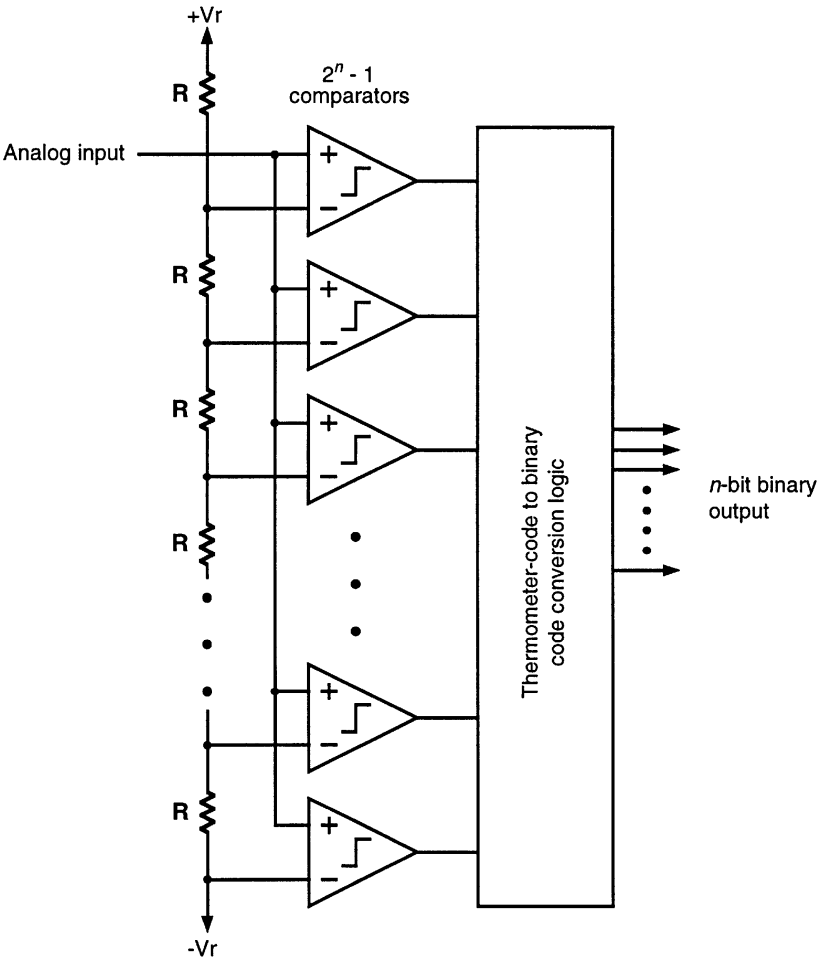


FIGURE 85.10 A flash converter has $2^n - 1$ comparators operating in parallel. It relies on the uniformity of the resistors for linearity.

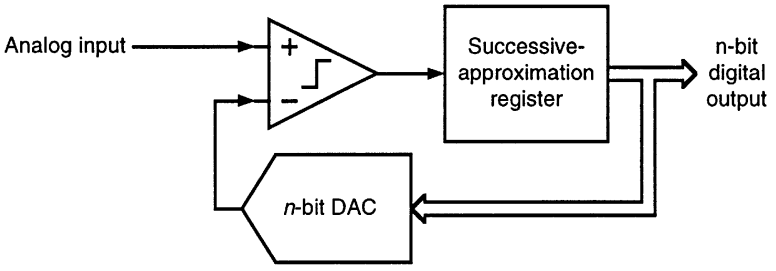


FIGURE 85.11 A successive-approximation converter has only one comparator and relies on an internal, precision DAC for linearity.

of iterations in a conversion. Instead of using just a comparator, the multistage ADC uses low-resolution flash converters (4 to 8 bits) as building blocks. Figure 85.13 illustrates an example of a 12-bit two-stage ADC built out of two flash ADCs and a fast DAC. The 6-bit flash ADC converts the residual error of the 8-bit flash ADC. The two digital outputs are combined to produce a 12-bit conversion result.

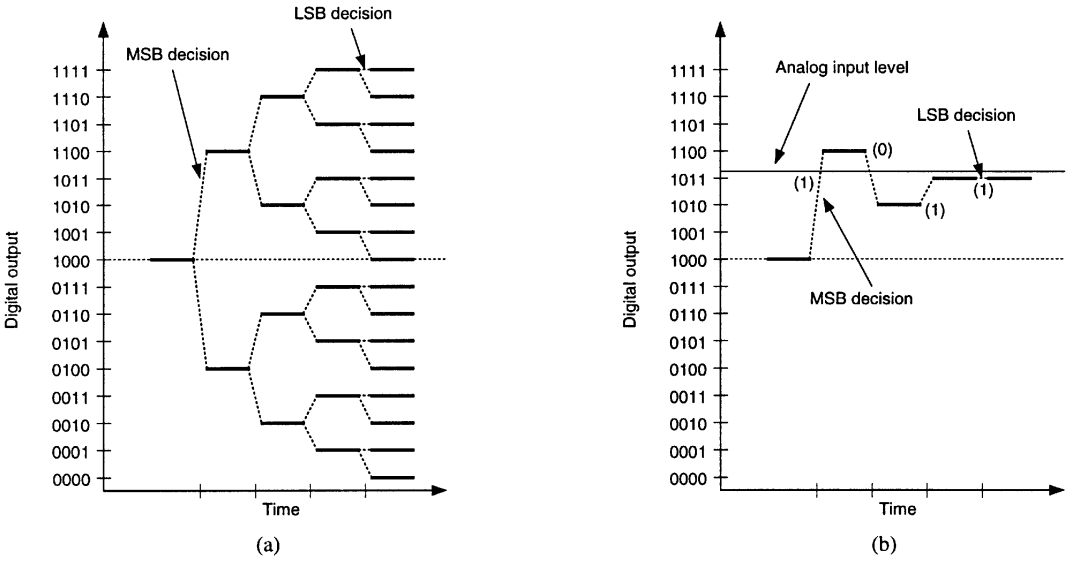


FIGURE 85.12 (a) Decision tree shows all the possible digital "guesses" of a four-bit successive-approximation converter over time. (b) Decision tree for conversion of four-bit code 1011.

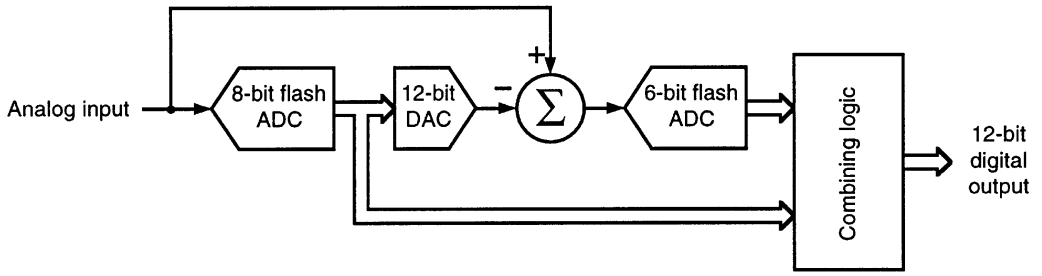


FIGURE 85.13 An example of a 12-bit multistage ADC built out of two flash ADCs and a fast DAC. The 8-bit flash ADC takes a first "guess" at the input signal and the 6-bit flash ADC converts the error in the guess, called the "residue." The 12-bit DAC actually needs to have only 8 bits, but it must be accurate to 12 bits. If the 8-bit flash ADC were perfect, the second flash ADC would only need 4 bits. But since the first flash actually may have some error, the second flash has 2 bits of "overlap."

If each flash ADC has a T/H at its input, then each stage can be converting the residual error from the previous stage while the previous stage is converting the next sample. The whole converter then can effectively operate at the sample rate of the slowest stage. Without the extra T/Hs, a new conversion cannot start until the residues have propagated through all the stages. This variation of the multistage ADC is called a *pipelined* ADC.

Integrating

Integrating converters are used for low-speed, high-resolution applications such as voltmeters. They are conceptually simple, consisting of an integrating amplifier, a comparator, a digital counter, and a very stable capacitor for accumulating charge (Figure 85.14). The most common integrating ADC in use is the dual-slope ADC, whose action is illustrated in Figure 85.15. Initially, the capacitor is discharged and so has no voltage across it. At time 0, the input to the integrator is switched to the analog input and the capacitor is allowed to charge for an amount of time, T1, which is always the same. Its rate of charging

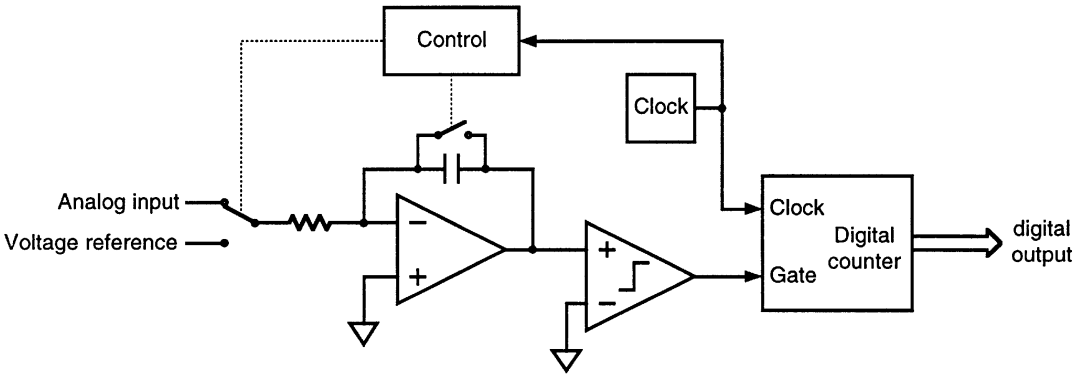


FIGURE 85.14 A dual-slope integrating converter uses a comparator to determine when the capacitor has fully discharged and relies on the capacitor for linearity.

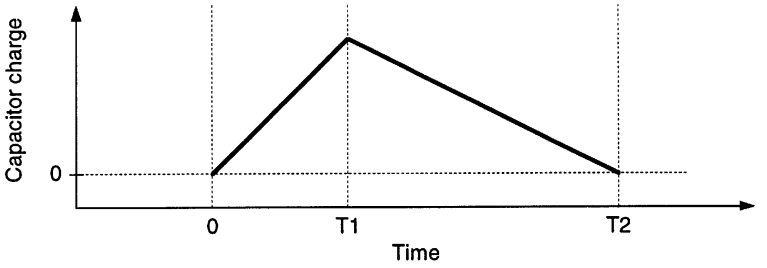


FIGURE 85.15 Charge on the integrating capacitor vs. time. At time 0, the input is switched to analog input and the switch across the capacitor opens. The capacitor integrates charge until fixed time T1. The input is then switched to the voltage reference to discharge the capacitor, and the counter begins counting a known clock. The comparator turns off the counter when the capacitor charge reaches 0 again, at time T2. The resulting count is proportional to the average input voltage over the time interval 0 to T1.

and thus its voltage at T1 are proportional to the input voltage. At time T1 the input switch flips over to the voltage reference, which has a negative value so that the capacitor will begin to discharge at a rate proportional to the reference. The counter measures how long it takes to discharge the capacitor completely. If the capacitor is of high quality, the ratio of the discharge time to the charge time is proportional to the ratio of the input voltage to the voltage reference, and so the counter output represents the analog input voltage.

An elaboration of the dual-slope ADC is the *multislope* integrating ADC. It achieves even higher resolution than the dual-slope ADC by discharging the capacitor at several progressively slower rates. At each rate, the counter is able to resolve finer increments of accumulated charge.

An important distinction between integrating converters and other ADCs is the way they sample the input voltage. Integrating converters do not sample the voltage itself; they *average* the voltage over the integration period and *then* they sample the average that is accumulated on the capacitor. This tends to reject noise that conventional sampling cannot, especially periodic noises. Most integrating ADCs operate with an integration period that is a multiple of one AC line period ($^1/_{60}$ or $^1/_{50}$ s) so that any potential interference from stray electric or magnetic fields caused by the power system is canceled.

Integrating converters are gradually being replaced in the marketplace with low-speed, high-resolution sigma–delta converters, which see. Sigma–delta converters are generally more flexible than integrating ADCs, and they are easier to use because they do not require an external charging capacitor. The resolution and speed of the two types are comparable, although integrating converters still have the highest linearity.

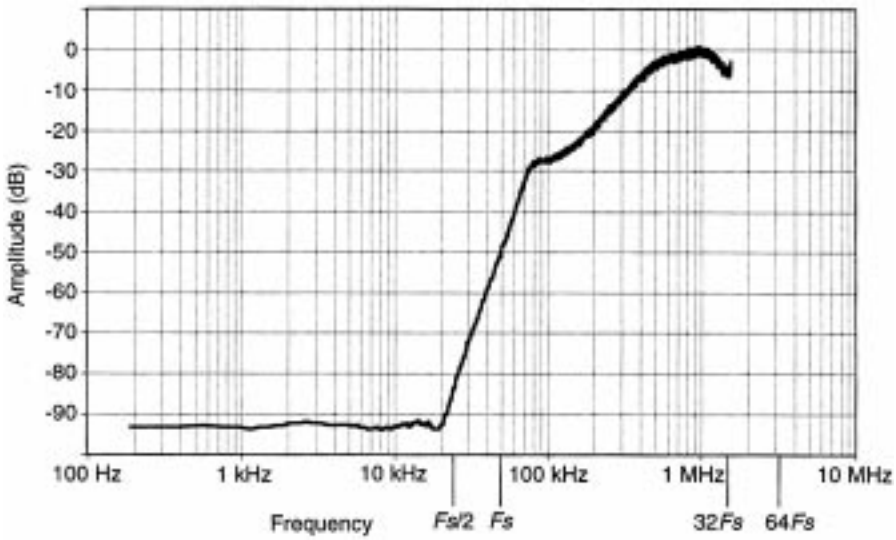


FIGURE 85.16 Spectrum of a 64-times oversampling SD ADC before the digital decimation filter. The modulator loop samples at 3.072 MS/s and the data comes out of the filter at 48 kS/s. The filter cuts off sharply at *Fs*/2, or 24 kHz, leaving only the small amount of noise left below 24 kHz.

Sigma–Delta ADCs

The *sigma–delta (SD)* ADC is quickly becoming one of the most popular types of ADC. DS ADCs typically have resolutions of 16 to 24 bits and sample rates of 100 kS/s down to 10 S/s. Because of their high resolution at 48 kS/s, they are the most common type of converters in modern digital audio equipment. DS ADCs defy intuition by quantizing initially with *very* low resolution (often one bit) at very high rates, typically 64× to 128× the eventual sample rate (called *oversampling*). The high-rate, low-resolution quantizer operates inside a feedback loop with an analog lowpass filter and a DAC to force the large amount of otherwise unavoidable quantization error (noise) to frequencies higher than the band of interest. The resulting spectral redistribution of the quantization noise is called *noise shaping*, illustrated in Figure 85.16. The low-resolution digital output of the ADC loop is fed into a digital filter that increases the resolution from the resolution of the ADC loop to the output resolution, reduces the data rate from the rate of the ADC loop to the output sample rate, and applies a low-pass digital filter, leaving only the signals in the frequency band of interest and a little inherent electronic noise.

Figure 85.17 shows how a one-bit sigma–delta ADC works. The comparator *is* the ADC, and its output is processed digitally, so that no further analog errors can accumulate. The comparator is in a feedback loop with a low-pass filter (typically third to fifth order) and a one-bit DAC. The one-bit DAC can take on only one of two values, +full-scale and –full-scale, so it is perfectly linear. The low-pass filter causes the loop gain to be high at low frequencies (the signal band of interest) and low at high frequencies. Since the error in a feedback loop is low when the gain is high and high when the gain is low, the errors dominate at high frequencies and are low in the band of interest. The result is a one-bit output whose duty cycle is proportional to the input signal. Together, the elements of the feedback loop are called a *sigma–delta modulator.*

Figure 85.18 illustrates the operation of a simple discrete-time (switched-capacitor) SD ADC. In this first-order example, the low-pass filter is just an integrator. The loop tries to force the input to the comparator back to the baseline, and the figure shows how the duty cycle of the resulting digital output reflects input signal. The digital data here have undesirable patterns which tend to repeat, called *limit cycles*. They can appear in the band of interest and interfere with the signal. Higher-order loop filters (third and above) make the bit activity so chaotic that it has no substantial limit cycles.

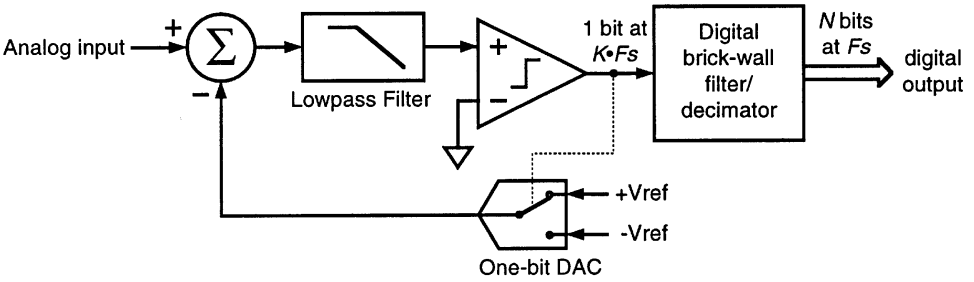


FIGURE 85.17 A SD modulating ADC uses a comparator simply as a one-bit quantizer. The linearity of a SD ADC is theoretically perfect because the one-bit DAC can only assume two values, and thus is linear by definition. Modern SD ADCs are made with switched-capacitor circuits which operate at *KFs*, where *Fs* is the output data sample rate and *K* is the oversampling ratio.

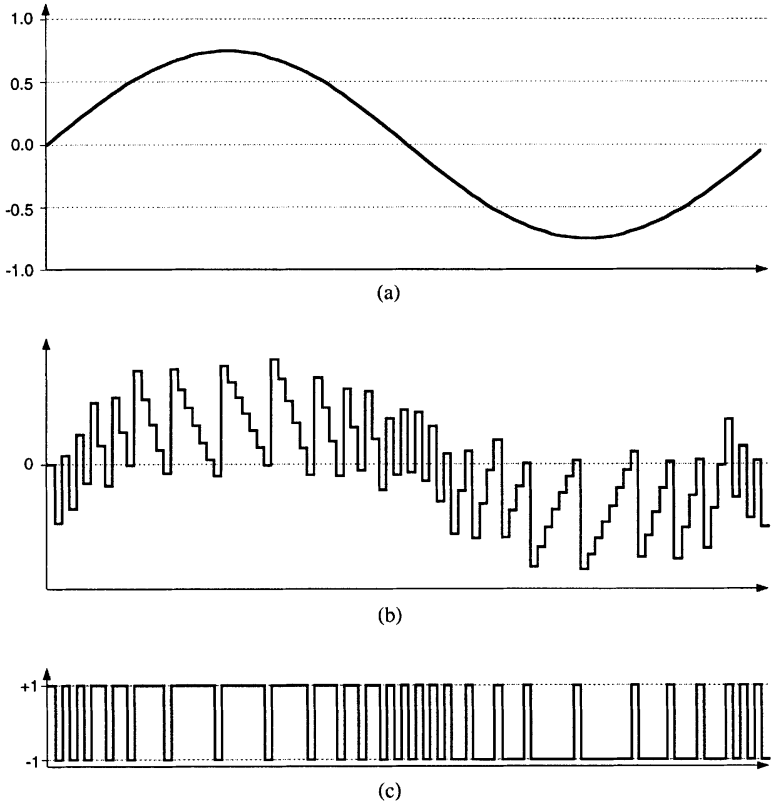


FIGURE 85.18 Behavior of a discrete-time (switched-capacitor) first-order SD modulator, where the low-pass filter is simply an integrator. In each graph, the *x*-axis represents time, and the *y*-axis represents signal level. (a) The input waveform. (b) Input to the comparator. (c) The one-bit digital comparator output. The duty cycle of this waveform corresponds to the input waveform. The digital filter and decimator recover the original waveform from this one bit.

The chief advantage of a SD converter is that it has a built-in antialias filter, and a darn good one at that. Most DS parts have a *finite-impulse response (FIR)* digital filter, which has an extremely flat frequency response in the passband and an extremely sharp cutoff, properties impossible to implement in analog

filters. The ADC still needs an antialias filter to reject signals above one half the oversampling rate. But this filter is simple to build, since it has to be flat only up to one half the output sampling rate and has many octaves (all the way to near the oversampling rate) to fall off. The combination of the two filters provides watertight protection from aliases, often 96 dB of attenuation over the entire spectrum.

An important improvement of the one-bit SD is the multibit SD, wherein the comparator is replaced by a flash converter with as much as four bits of resolution. This improves the ENOB of the whole converter by several bits.

Voltage-to-Frequency Converters

Voltage-to-frequency converters (VFCs) are versatile, low-cost circuits that convert analog voltages to periodic waveforms whose frequency is proportional to the analog input voltage. A VFC is conceptually similar to an integrating converter (see above) except that the digital counter is missing and is replaced with a short-pulse generator that quickly discharges the capacitor. The voltage reference is not connected intermittently to the input; instead, it appears all the time at the minus input of the comparator instead of ground. The capacitor charges at a rate proportional to the input voltage until the voltage is equal to the voltage reference. Then the comparator trips the pulse generator, which quickly discharges the capacitor, and the cycle begins again. The periodic pulse at the comparator output can be used as the digital output.

The advantage of the VFC over conventional ADCs is that the one-bit output can be transmitted digitally, through isolation transformers, through fiber-optic cable, or through any other isolating, nonisolating, long-distance, or short-distance transmission medium. All that is needed at the receiving end to complete the analog-to-digital conversion is a digital counter, which does not need to be synchronized to the VFC itself. Sometimes, the digital conversion is not needed; a VFC can be used with an isolating transformer and a *frequency-to-voltage converter (FVC)* to create an isolation amplifier. For a good discussion of VFCs, see Reference 7.

85.6 Instrumentation and Components

Integrated Circuits

Table 85.1 lists several popular high-quality ADCs in integrated circuit form. The prices given are approximate for small quantities and for the lowest grade of part, as of mid-1996. By no means exhaustive, the list is sampling of a few of the most popular or best-performing chips of each type of ADC. Table 85.2 contains addresses, phone numbers, and internet sites for the manufacturers in Table 85.1.

Instrumentation

Plug-in data acquisition cards are becoming increasingly popular as personal computer prices come down and processor performance goes up. These cards typically contain one or more ADCs (with S/H), instrumentation amplifiers with gain and differential input, and multiplexers to switch to different inputs. Some have DACs on-board, and some have digital data and timing functions as well. Once considered low performance and hard to use, data acquisition cards have improved dramatically, equaling and in some cases exceeding capabilities of stand-alone instruments. Most come with drivers that interface to user-friendly software packages for creating easy-to-use yet custom-built computer instrumentation. Table 85.3 lists a few popular plug-in data acquisition boards and Table 85.4 lists how their manufacturers may be contacted.

TABLE 85.1 ADC Integrated Circuits

Part	Type	Sample Rate	Resolution, bits	Manufacturer	Approx. Price, $
ADC160	Integrating	1 S/s	24	Thaler	225.00
AD7714	Sigma–delta	2.62 S/s	24	Analog Devices	22.00
MAX132	Integrating	6 S/s	19	MAXIM	15.11
CS5508	Sigma–delta	20 S/s	20	Crystal	21.50
HI7190	Sigma–delta	10 S/s	24	Harris	17.85
AD1879	Sigma–delta	50 kS/s	18	Analog Devices	46.00
CS5390	Sigma–delta	50 kS/s	20	Crystal	75.30
ADS7809	SAR	100 kS/s	16	Burr-Brown	41.54
CS5101A	SAR	100 kS/s	16	Crystal	67.20
AD7893	SAR	117 kS/s	12	Analog Devices	14.00
AD976	SAR	200 kS/s	16	Analog Devices	36.50
AD7722	Sigma–delta	200 kS/s	16	Analog Devices	39.80
LTC1278	SAR	500 kS/s	12	Linear Technology	17.08
AD1385	Multistage	500 kS/s	16	Analog Devices	1053.00
ADS7819	SAR	800 kS/s	12	Burr-Brown	31.90
AD9220	Multistage	10 MS/s	12	Analog Devices	22.95
AD775	Multistage	20 MS/s	8	Analog Devices	14.00
AD9050	Multistage	40 MS/s	10	Analog Devices	39.00
AD9066	Flash	60 MS/s	6	Analog Devices	7.00
HI1276	Flash	500 MS/s	8	Harris	338.58

TABLE 85.2 Companies That Manufacture ADC Integrated Circuits

Analog Devices, Inc.
One Technology Way
P.O. Box 9106
Norwood, MA 02062-9106
(617) 329-4700
http://www.analog.com

Burr-Brown Corporation
P.O. Box 11400
Tucson, AZ 85734-1400
(520) 746-1111
http://www.burr-brown.com

Crystal Semiconductor Corporation
P.O. Box 17847
Austin, TX 78760
(512) 445-7222
http://www.cirrus.com/prodtech/crystal.html

Harris Corp. Semiconductor
Products Division
P.O. Box 883
Melbourne, FL 37902
(407) 729-4984
http://www.semi.harris.com

Linear Technology Corporation
1630 McCarthy Blvd.
Milpitas, CA 95035-7417
(408) 432-1900
http://www.linear-tech.com

Maxim Integrated Products, Inc.
120 San Gabriel Drive
Sunnyvale, CA 94086
(408) 737-7600
http://www.maxim-ic.com

Thaler Corporation
2015 N. Forbes Boulevard
Tucson, AZ 85745
(520) 882-4000
http://www.thaler.com

TABLE 85.3 Plug-In Data Acquisition Boards

Part	Type	Sample Rate	Resolution, bits	Manufacturer	Approx. Price, $
AT-A2150	Sigma–delta	51.2 kS/s	16	National Instruments	1495
AT-MIO-16XE-50	SAR	20 kS/s	16	National Instruments	995
AT-MIO-16E-10	SAR	100 kS/s	12	National Instruments	995
CIO-DAS1600/12	SAR	160 kS/s	12	ComputerBoards, Inc.	599
AT-MIO-16XE-10	SAR	100 kS/s	16	National Instruments	1995
CIO-DAS1600/16	SAR	100 kS/s	16	ComputerBoards, Inc.	699
DT-3001	SAR	330 kS/s	12	Data Translation, Inc.	995
DAS-1800AO	SAR	333 kS/s	12	Keithley Metrabyte	1299
AT-MIO-16E-1	SAR	1 MS/s	12	National Instruments	1795
FAST16-1	Multistage	1 MS/s	16	Analogic	3895

TABLE 85.4 Companies That Manufacture Plug-In Data Acquisition Boards

Analogic Corporation 360 Audubon Road Wakefield, MA 01880 (508) 977-3000	Keithley Metrabyte 440 Myles Standish Blvd. Taunton MA 02780 (508) 880-3000 http://www.metrabyte.com
ComputerBoards, Inc. 125 High Street Mansfield, MA 02048 (508) 261-1123	National Instruments Corporation 6504 Bridge Point Parkway Austin, TX 78730 (512) 794-0100 http://www.natinst.com
Data Translation, Inc. 100 Locke Drive Marlboro, MA 01752-1192 (508) 481-3700 http://www.datx.com	

References

1. C. E. Shannon, Communication in the presence of noise, *Proc. IRE*, 37(1): 10–21, 1949.
2. H. Nyquist, Certain topics in telegraph transmission theory, *AIEE Transactions*, 617–644, April, 1928.
3. A. B. Williams and F. J. Taylor, *Electronic Filter Design Handbook: LC, Active, and Digital Filters*, 2nd ed., New York: McGraw-Hill, 1988.
4. S. P. Lipshitz, R. A. Wannamaker, and J. Vanderkooy, Quantization and dither: a theoretical survey, *J. Audio Eng. Soc.*, 40, 355–375, 1992.
5. H. W. Ott, *Noise Reduction Techniques in Electronic Systems*, 2nd ed., New York: John Wiley & Sons, 1988.
6. R. Morrison, *Grounding and Shielding Techniques in Instrumentation*, 3rd ed., New York: John Wiley & Sons, 1986.
7. J. Williams, Designs for High Performance Voltage-to-Frequency Converters, Application Note 14, Linear Technology Corporation, March 1986.

Further Information

M. Demler, *High-Speed Analog-to-Digital Conversion*, San Diego, CA: Academic Press, 1991.

B. M. Gordon, *The Analogic Data-Conversion Systems Digest*, Wakefield, MA: Analogic Corporation, 1981.

D. H. Sheingold, Ed., *Analog-Digital Conversion Handbook*, Englewood Cliffs, NJ: Prentice-Hall, 1986.

Crystal Semiconductor, *Delta Sigma A/D Conversion Technique Overview*, Application Note AN10, Austin, TX, 1989.