SECTION 5

DATA CONVERTERS

- DC Errors in Data Converters
- AC Errors in Data Converters
- AC Errors in DACs
- AC Errors in ADCs
- Components and Processes for Data Converters
- DAC Structures
- DAC Logic
- ADC Structures:
  Comparators, Flash or Parallel ADCs, Subranging ADCs, Integrating ADCs, Voltage-to-Frequency Converter/Counter ADCs, Tracking ADCs, Successive Approximation ADCs, Sigma-Delta ADCs, Sample-and-Hold Amplifiers (SHAs)

SECTION 5

DATA CONVERTERS

James Bryant, Walt Kester

INTRODUCTION

This section is intended as an introduction to the techniques used in the design of digital-to-analog and analog-to-digital converters (DACs and ADCs, respectively). There is a common misconception that as digital computing becomes more ubiquitous and less expensive, analog circuitry will soon disappear altogether. In fact, nothing could be further from the truth - the entire universe above the quantum level is analog, and the more ubiquitous digital computers become, the more analog transducers and circuitry will be necessary to interface them to the real world.

DACs and ADCs are these interfaces between the analog and digital worlds. A DAC is a device which gives an analog output related to a digital input, while an ADC gives a digital output in response to an analog input (see Figure 5.1). While DACs and ADCs do exist which have intentionally nonlinear transfer characteristics, they are not common, and will not be considered here. If we disregard such nonlinear exceptions, we can say that a DAC is a circuit whose analog output is proportional to both its analog reference and to the value of its digital input. Conversely, an ADC is a circuit whose digital output is proportional to the ratio of its analog input to its analog reference. (Often, but by no means always, the scaling factor between the analog reference and the analog signal is unity, so the digital signal represents the normalized ratio of the two.)

DIGITAL-TO-ANALOG AND ANALOG-TO-DIGITAL CONVERTERS

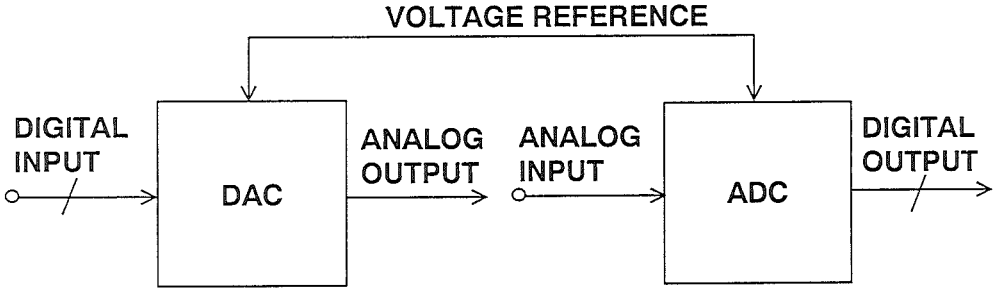


Figure 5.1

The most important thing to remember about both DACs and ADCs is that one of the signals is digital, and therefore quantized. That is, an N-bit word represents one of 2^N possible states, and therefore an N-bit DAC (with a fixed reference) can have only 2^N possible analog outputs, and an N-bit ADC can have only 2^N possible digital outputs.

The analog signals will generally be voltages or currents, but may be resistance, and can be any analog quantity, not necessarily an electrical one. Such cases, where a transducer is integrated with a data converter, are less common, but certainly not unknown. The most common are optical: DACs with light output and ADCs with integral photocells.

The resolution of data converters may be expressed in several different ways: The weight of the Least Significant Bit (LSB), parts per million of full scale (ppm FS), millivolts (mV), etc. Different devices (even from the same manufacturer) will be specified differently, so converter users must learn to translate between the different types of specifications if they are to compare devices successfully. The size of the least significant bit for various resolutions is shown in Figure 5.2.

THE SIZE OF A LEAST SIGNIFICANT BIT (LSB)

RESOLUTION N	2^N	VOLTAGE (10V FS)	ppm FS	% FS	dB FS
2-bit	4	2.5 V	250,000	25	–12
4-bit	16	625 mV	62,500	6.25	–24
6-bit	64	156 mV	15,625	1.56	–36
8-bit	256	39.1 mV	3,906	0.39	–48
10-bit	1,024	9.77 mV (10 mV)	977	0.098	–60
12-bit	4,096	2.44 mV	244	0.024	–72
14-bit	16,384	610 µV	61	0.0061	–84
16-bit	65,536	153 µV	15	0.0015	–96
18-bit	262,144	38 µV	4	0.0004	–108
20-bit	1,048,576	9.54 µV (10 µV)	1	0.0001	–120
22-bit	4,194,304	2.38 µV	0.24	0.000024	–132
24-bit	16,777,216	596 nV*	0.06	0.000006	–144

* 600nV is the Johnson Noise in a 10kHz BW of a 2.2kΩ Resistor @ 25°C

Remember: 10-bits and 10V FS yields an LSB of 10mV, 1000ppm, or 0.1%
All other values may be calculated by powers of 2

Figure 5.2

Before we can consider the various architectures used in data converters, it is necessary to consider the performance to be expected, and the specifications which are important. The next three parts of this section will consider the definitions of errors and specifications used for data converters, so that when we consider particular architectures, we can understand their strengths and weaknesses.

DC ERRORS IN DATA CONVERTERS

The first applications of data converters were in measurement and control applications where the exact timing of the conversion was usually unimportant, and the data rate was slow. In such applications, the DC specifications of converters are important, but timing and AC specifications are not. Today many, if not most, converters are used in "sampling" and "reconstruction" systems where AC specifications are critical (and DC ones may not be) - these will be considered in the next part of this section.

Figure 5.3 shows the ideal transfer characteristics for a 3-bit unipolar DAC, and Figure 5.4 a 3-bit unipolar ADC. In a DAC, both the input and the output are quantized, and the graph consists of eight points - while it is reasonable to discuss the line through these points, it is very important to remember that the actual transfer characteristic is *not* a line, but a number of discrete points.

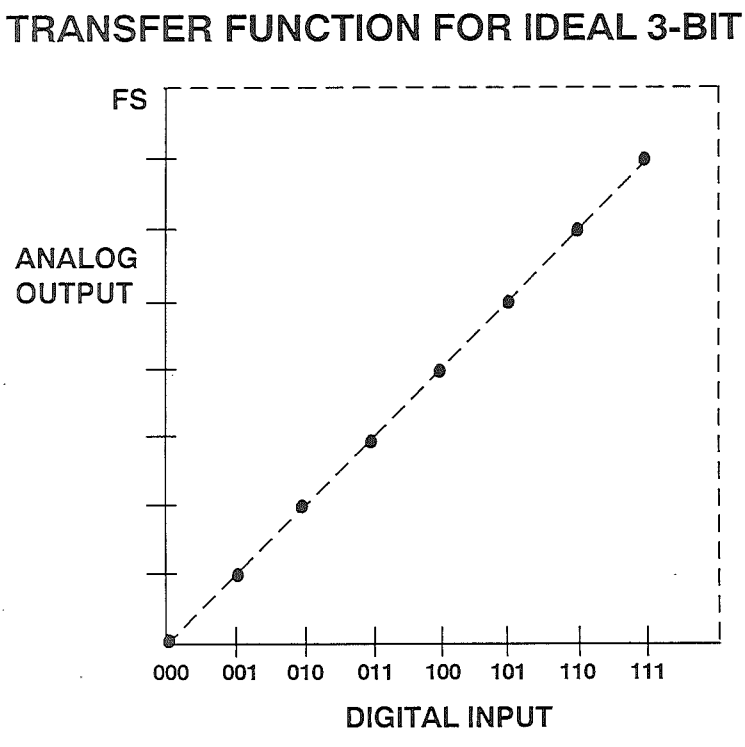


Figure 5.3

TRANSFER FUNCTION FOR IDEAL 3-BIT ADC

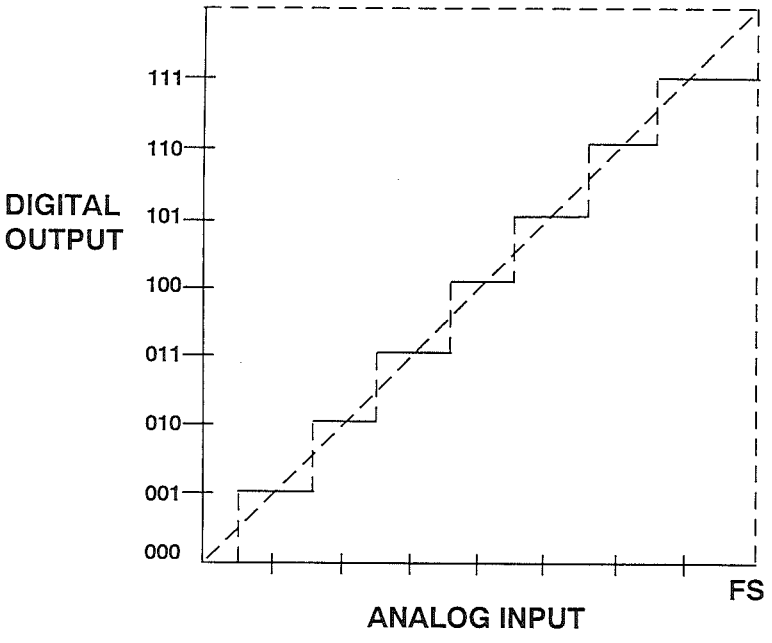


Figure 5.4

The input to an ADC is analog and is not quantized, but its output is quantized. The transfer characteristic therefore consists of eight horizontal steps (when considering the offset, gain and linearity of an ADC we consider the line joining the midpoints of these steps).

In both cases, digital full scale (all "1"s) corresponds to 1 LSB below the analog full scale (the reference, or some multiple thereof). This is because, as mentioned above, the digital code represents the *normalized* ratio of the analog signal to the reference. If this were unity, the digital code would be all "0"s and "1" in the bit *above* the Most Significant Bit (MSB).

The (ideal) ADC transitions take place at ½ LSB above zero, and thereafter every LSB, until 1½ LSB below analog full scale. Since the analog input to an ADC can take any value, but the digital output is quantized, there may be a difference of up to ½ LSB between the actual analog input and the exact value of the digital output. This is known as the "quantization error" or "quantization uncertainty" as shown in Figure 5.5. In AC (sampling) applications this quantization error gives rise to "quantization noise" which will be discussed in the next section.

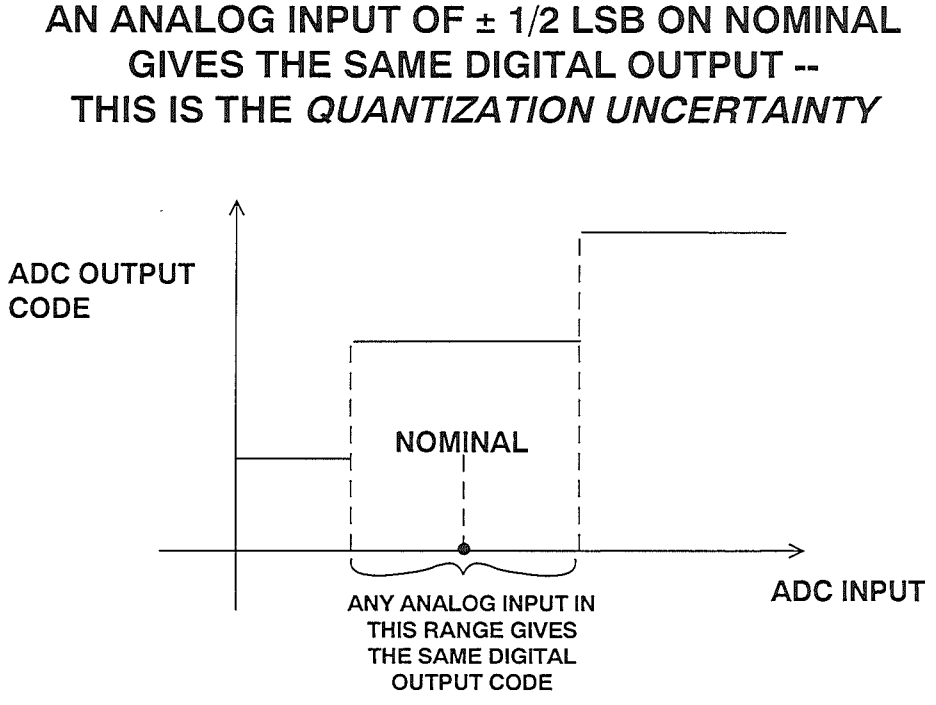


Figure 5.5

There are many possible digital coding schemes for data converters: *binary*, *offset binary*, *1's complement*, *2's complement*, *gray code*, *BCD* and others. This section, being devoted mainly to the *analog* issues surrounding data converters, will use simple *binary* and *offset binary* in its examples and will not consider the merits and disadvantages of these, or any other, forms of digital code.

The examples in Figures 5.3 and 5.4 use unipolar converters, whose analog port has only a single polarity. These are the simplest type, but bipolar converters are more generally useful. There are two types of bipolar converter: the simpler is merely a unipolar converter with an accurate 1 MSB of negative offset (and many converters are arranged so that this offset may be switched in and out so that they can be used as either unipolar or bipolar converters at will), but the other, known as a *sign-magnitude* converter is more complex, and has N bits of magnitude information and an additional bit which corresponds to the sign of the analog signal. Sign-magnitude DACs are quite rare, and sign magnitude ADCs are found mostly in digital voltmeters (DVMs). Figure 2.6 shows the transfer functions for these unipolar, offset bipolar, and sign-magnitude bipolar codes.

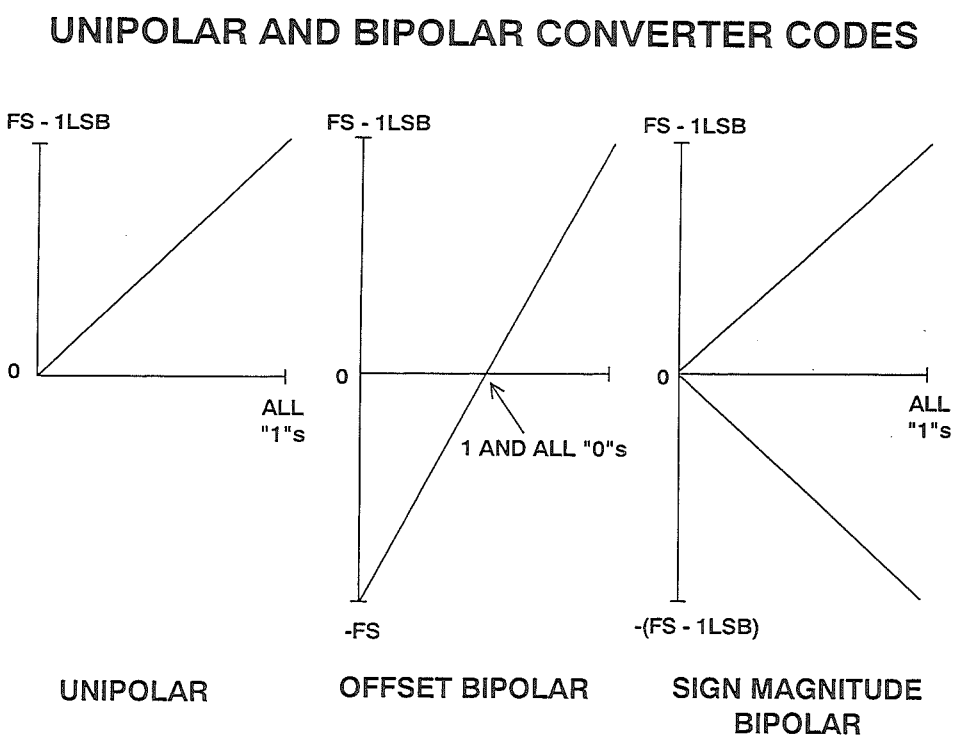


Figure 5.6

The four DC errors in a data converter are offset error, gain error, and two types of linearity error. Offset and gain errors are analogous to offset and gain errors in amplifiers (see Figure 5.7). (Though offset error and zero error, which are identical in amplifiers and unipolar data converters, are not identical in bipolar converters and should be carefully distinguished.) The transfer characteristics of both DACs and ADCs may be expressed as D = K + GA, where D is the digital code, A is the analog signal, and K and G are constants. In a unipolar converter, K is zero, and in an offset bipolar converter, it is –1 MSB.

The offset error is the amount by which the actual value of K differs from its ideal value. The gain error is the amount by which G differs from its ideal value, and is generally expressed as the percentage difference between the two, although it may be defined as the gain error contribution (in mV or LSB) to the total error at full scale. These errors can usually be trimmed by the data converter user. Note, however, that amplifier offset is trimmed at zero input, and then the gain is trimmed near to full scale. The trim algorithm for a bipolar data converter is not so straightforward.

CONVERTER OFFSET AND GAIN ERROR

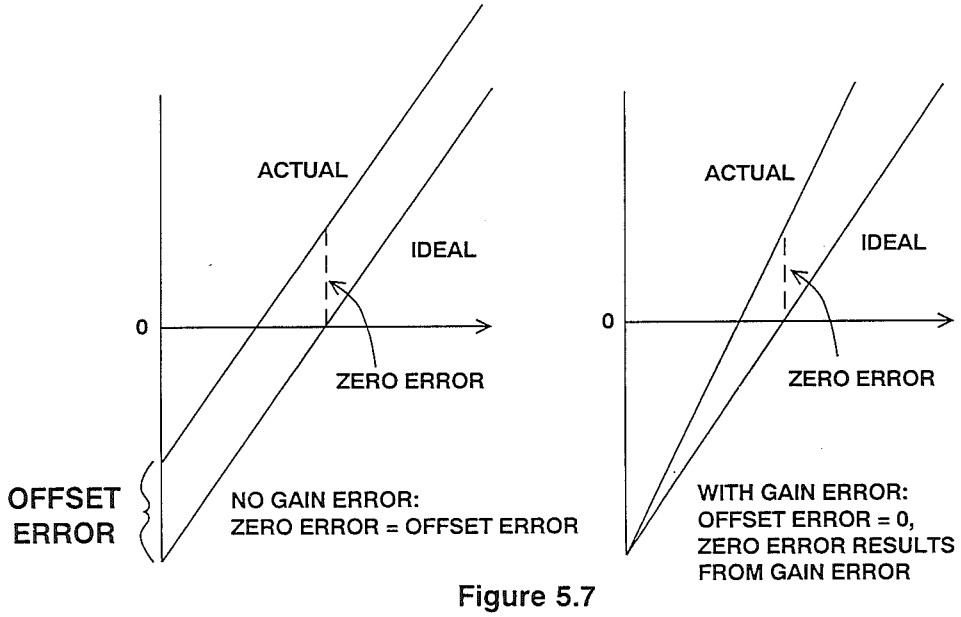


Figure 5.7

The integral linearity error of a converter is also analogous to the linearity error of an amplifier, and is defined as the maximum deviation of the actual transfer characteristic of the converter from a straight line, and is generally expressed as a percentage of full scale (but may be given in LSBs). There are two common ways of choosing the straight line: *end point* and *best straight line* (see Figure 5.8).

METHOD OF MEASURING INTEGRAL LINEARITY ERRORS (SAME CONVERTER ON BOTH GRAPHS)

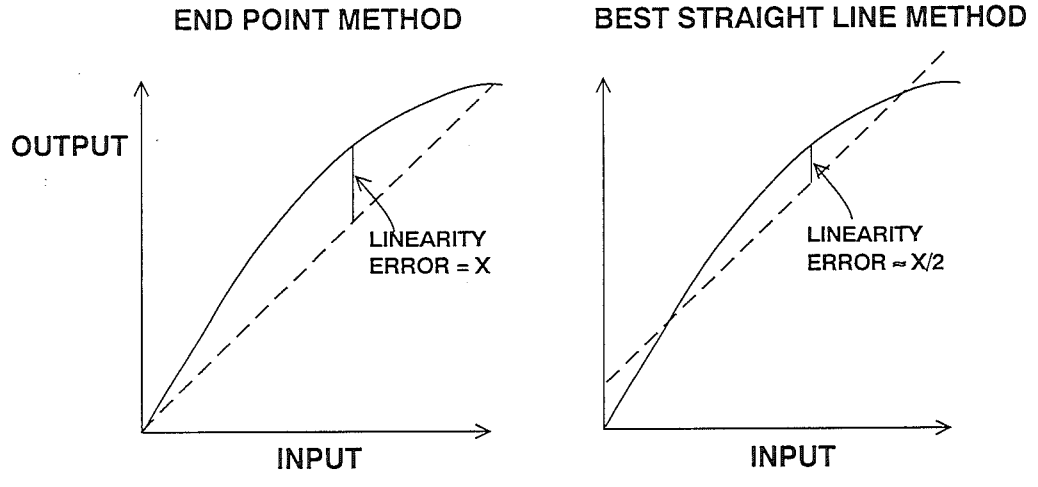


Figure 5.8

In the end point system, the deviation is measured from the straight line through the origin and the full scale point (after gain adjustment). This is the most useful integral linearity measurement for measurement and control applications of data converters (since error budgets depend on deviation from the ideal transfer characteristic, not from some arbitrary "best fit"), and is the one normally adopted by Analog Devices Inc.

The best straight line, however, does give a better prediction of distortion in AC applications, and also gives a lower value of "linearity error" on a data sheet. The best fit straight line is drawn through the transfer characteristic of the device using standard curve fitting techniques, and the maximum deviation is measured from this line. In general, the integral linearity error measured in this way is only 50% of the value measured by end point methods. This makes the method good for producing impressive data sheets, but it is less useful for error budget analysis. For AC applications, it is even better to specify distortion than DC linearity, so it is rarely necessary to use the best straight line method to define converter linearity. Analog Devices uses this definition on data sheets only in the case of second-source products where the original manufacturer's data sheet used best straight line, rather than end point, linearity.

The other type of converter non-linearity is "differential non-linearity" (DNL). This relates to the linearity of the code transitions of the converter. In the ideal case, a change of 1 LSB in digital code corresponds to a change of exactly 1 LSB of analog signal. In a DAC, a change of 1 LSB in digital code produces exactly 1 LSB change of analog output, while in an ADC there should be exactly 1 LSB change of analog input to move from one digital transition to the next.

Where the change in analog signal corresponding to 1 LSB digital change is more or less than 1 LSB, there is said to be a DNL error. The DNL error of a converter is normally defined as the maximum value of DNL to be found at any transition.

If the DNL of a DAC is less than –1 at any transition (see Figure 5.9), the DAC is "non-monotonic" i.e., its transfer characteristic contains one or more maxima or minima. A DNL greater than +1 does not cause non-monotonicity, but is still undesirable. In many DAC applications (especially closed-loop systems where non-monotonicity can change negative feedback to positive feedback), it is critically important that DACs are monotonic. DAC monotonicity is often explicitly specified on data sheets, although if the DNL is guaranteed to be less than 1 LSB (i.e., $|DNL| \leq 1LSB$) then the device must be monotonic, even without an explicit guarantee.

ADCs can be non-monotonic, but a more common result of excess DNL in ADCs is missing codes (see Figure 5.10). Missing codes (or non-monotonicity) in an ADC are as objectionable as non-monotonicity in a DAC. Again, they result from DNL > 1 LSB.

TRANSFER FUNCTION OF NON-IDEAL 3-BIT DAC SHOWS DIFFERENTIAL NON-LINEARITY AND NON-MONOTONICITY

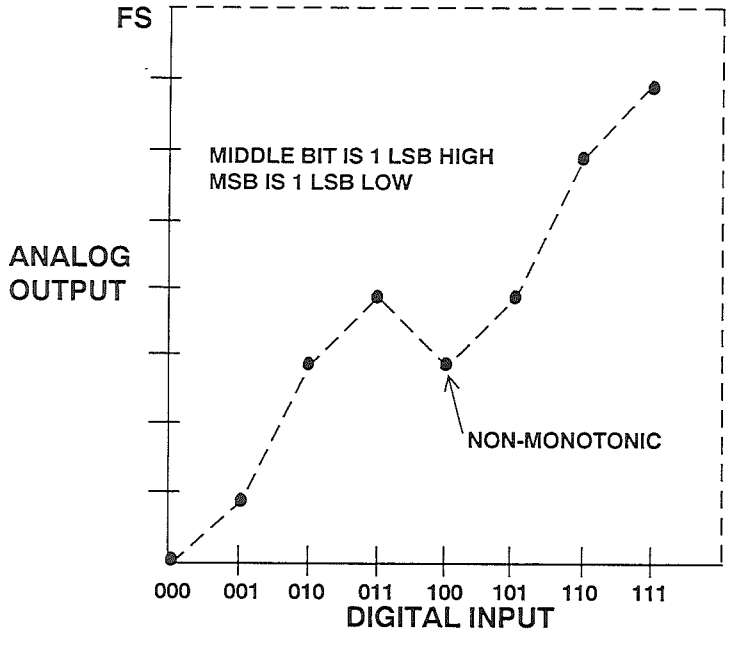


Figure 5.9

TRANSFER FUNCTION OF NON-IDEAL 3-BIT ADC SHOWS DIFFERENTIAL NON-LINEARITY AND MISSING CODE

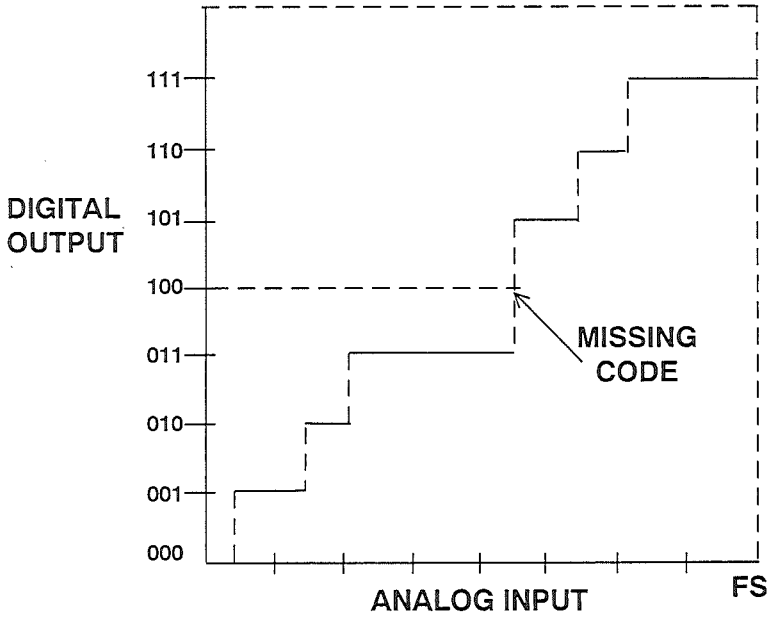


Figure 5.10

Defining missing codes is more difficult than defining non-monotonicity. All ADCs suffer from some transition noise as shown in Figure 5.11 (think of it as the flicker between adjacent values of the last digit of a DVM). As resolutions become higher, the range of input over which transition noise occurs may approach, or even exceed, 1 LSB. In such a case, especially if combined with a negative DNL error, it may be that there are some (or even all) codes where transition noise is present for the whole range of inputs. There are therefore some codes for which there is no input which will guarantee that code as an output, although there may be a range of inputs which will sometimes produce that code.

COMBINED EFFECTS OF ADC CODE TRANSITION NOISE AND DNL

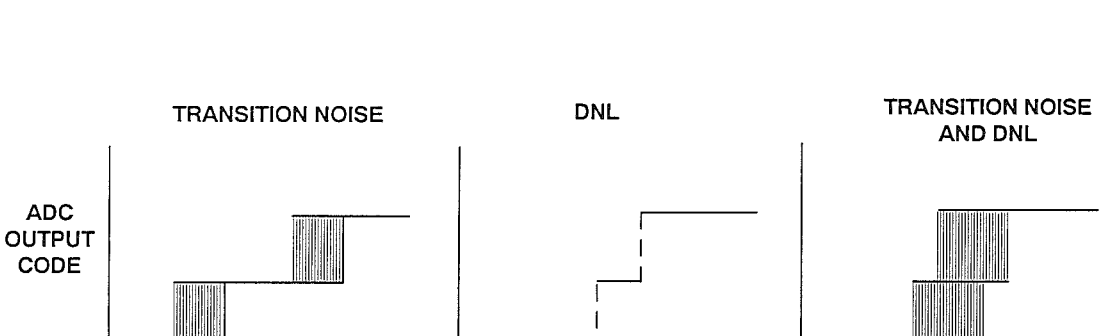


Figure 5.11

For lower resolution ADCs, it may be reasonable to define "no missing codes" as a combination of transition noise and DNL which guarantees some level (perhaps 0.2 LSB) of noise-free code for all codes. However, this is impossible to achieve at the very high resolutions achieved by modern sigma-delta ADCs, or even at lower resolutions in wide bandwidth sampling ADCs. In these cases, the manufacturer must define noise levels and resolution in some other way. Which method is used is less important, but the data sheet should contain a clear definition of the method used and the performance to be expected.

AC ERRORS IN DATA CONVERTERS

Over the last decade, the major application of data converters has shifted from DC measurement and control applications to AC sampling and reconstruction. Detailed descriptions and definitions of sampled data systems are beyond the scope of this section, but they are thoroughly discussed in Reference 1 and in Section 4 of this book. In very simple terms, a "sampled data system" is a system where the instantaneous value of an AC waveform is sampled at regular intervals (see Figure 5.12). The resulting digital codes may be used to store the waveform (as in CDs and DATs), or intensive computation on the samples (Digital Signal Processing, or DSP) may be used to perform filtering, compression, and other operations. The inverse operation, "reconstruction", occurs when a series of digital codes are fed to a DAC to reconstruct an AC waveform - an obvious example of this is a CD or DAT player, but the technique is very widely used indeed in telecommunications, radio, synthesizers, and many other applications.

SAMPLED AND RECONSTRUCTED WAVEFORMS

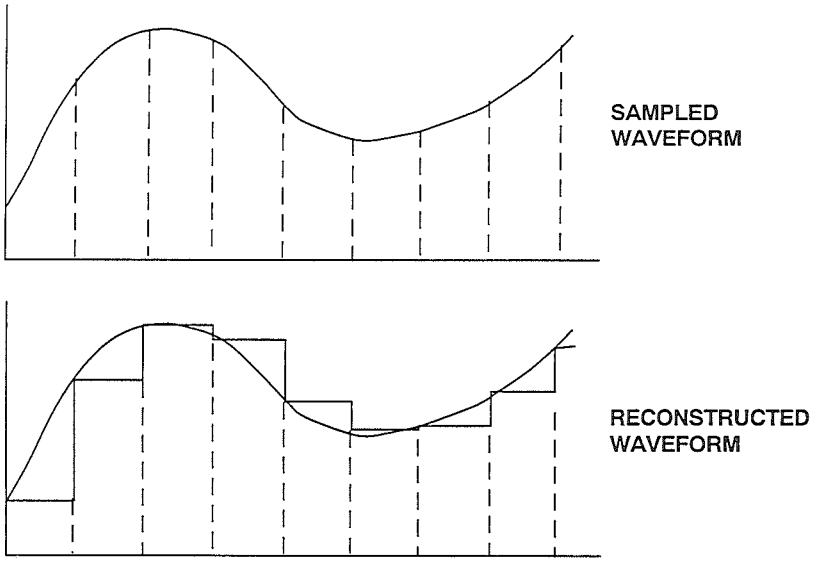


Figure 5.12

The data converters used in these applications must have good performance with AC signals, but may not require good DC specifications. The first high performance converters to be designed for such applications were often manufactured with good AC specifications but poor, or unspecified, DC performance. Today the design tradeoffs are better understood, and most converters will have good, and guaranteed, AC and DC specifications. DACs for digital audio, however, which must be extremely competitive in price, are generally sold with comparatively poor DC specifications — not because their DC performance is poor, but because it is not tested during manufacture.

While it is easier to discuss the DC parameters of both DACs and ADCs together, their AC specifications are sufficiently different to deserve separate consideration.

AC ERRORS IN DACs

The AC specifications which are most likely to be important with DACs are *settling time, glitch, distortion, Spurious Free Dynamic Range (SFDR),* and *phase noise.*

The settling time of a DAC is the time from a change of digital code to when the output comes within *and remains within* some error band as shown in Figure 5.13. With amplifiers, it is hard to make comparisons of settling time, since their error bands may differ from amplifier to amplifier, but with DACs the error band will almost invariably be ±1 or ±½ LSB.

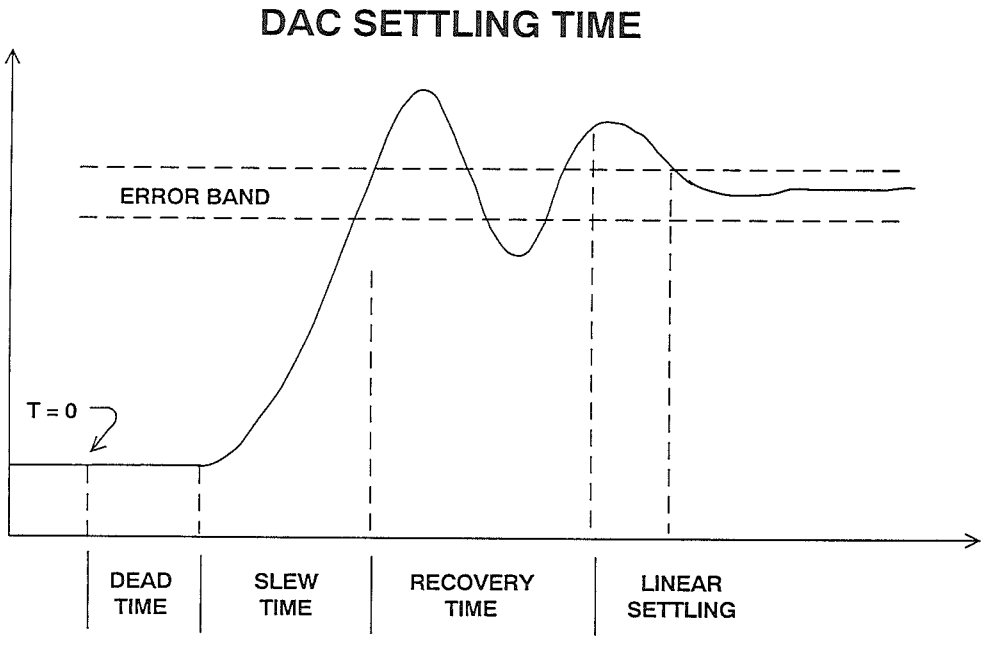


Figure 5.13

The settling time of a DAC is made up of four different periods: the *switching time* or *dead time* (during which the digital switching, but not the output, is changing), the *slewing time* (during which the rate of change of output is limited by the slew rate of the DAC output), the *recovery time* (when the DAC is recovering from its fast slew and may overshoot), and the *linear settling time* (when the DAC output approaches its final value in an exponential or near-exponential manner). If the slew time is short compared to the other three (as is usually the case with current output DACs), then the settling time will be largely independent of the output step size. On the other hand, if the slew time is a significant part of the total, then the larger the step, the longer the settling time.

Ideally, when a DAC output changes it should move from one value to its new one monotonically. In practice, the output is likely to overshoot, undershoot, or both (see Figure 5.14). This uncontrolled movement of the DAC output during a transition is known as *glitch*. It can arise from two mechanisms: capacitive coupling of digital transitions to the analog output, and the effects of some switches in the DAC operating more quickly than others and producing temporary spurious outputs.

DAC TRANSITIONS (SHOWING GLITCH)

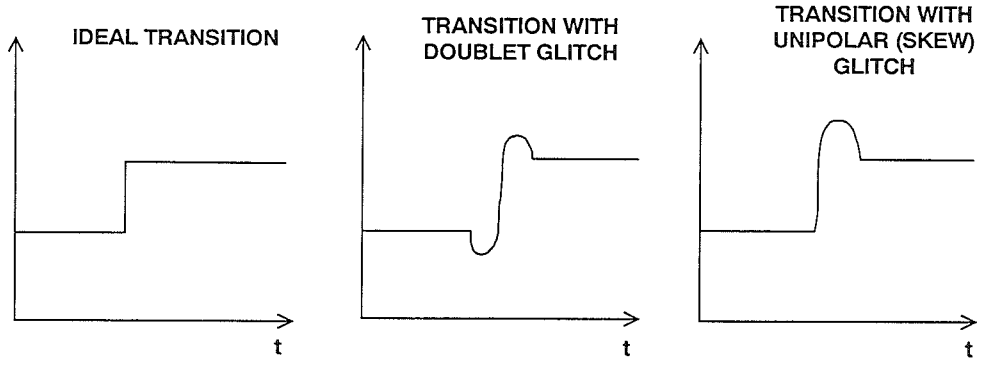


Figure 5.14

Capacitive coupling frequently produces roughly equal positive and negative spikes (sometimes called a *doublet* glitch) which more or less cancel in the longer term. These glitches occur whether or not a reference voltage is present on the DAC. The glitch produced by switch timing differences is generally unipolar, much larger, and does not occur in the absence of a reference (so the effects of capacitive coupling and switch timing may often be observed separately).

Glitch is generally (inaccurately) defined in terms of *glitch energy*. The term is a misnomer, since the unit is Volt-seconds (or more probably μV-sec or pV-sec), and the magnitude of a glitch is better referred to as the *glitch impulse area*, or simply, *glitch impulse*. The midscale glitch produced by the transition between the codes 1000...000 and 0111...111 is usually the worst glitch. Glitches at other code transition points (such as 1/4 and 3/4 full scale) are generally less. Figure 5.15 shows the midscale glitch for a fast low-glitch DAC.

AD9720/AD9721 DAC MIDSCALE GLITCH SHOWS 1.34pV-S NET IMPULSE AREA AND SETTLING TIME OF 4.5ns

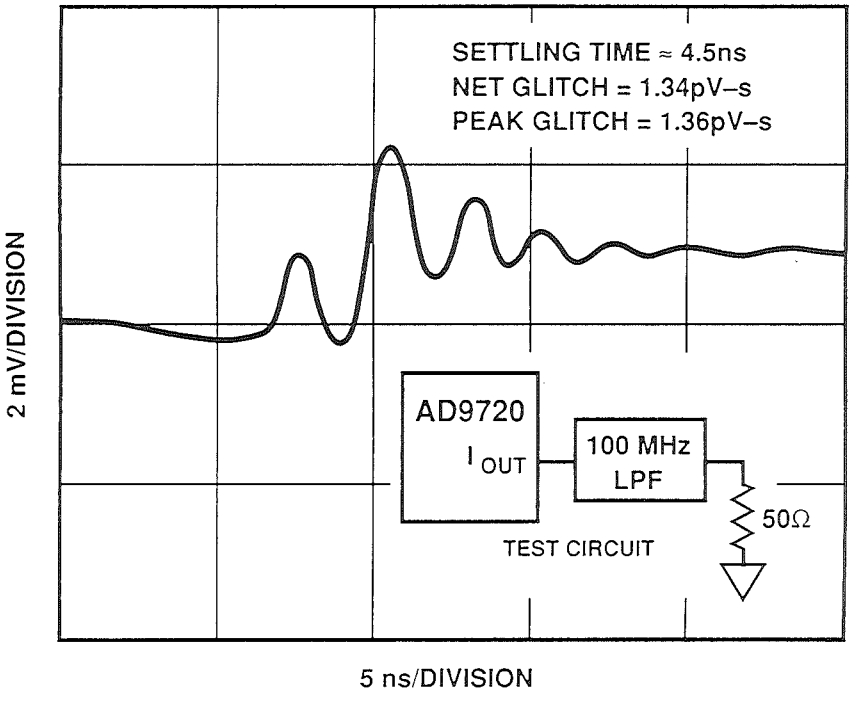


Figure 5.15

If we consider the spectrum of a waveform reconstructed by a DAC from digital data, we find that in addition to the expected spectrum (which will contain one or more frequencies, depending on the nature of the reconstructed waveform), there will also be noise and distortion products. Distortion may be specified in terms of harmonic distortion, Spurious Free Dynamic Range (SFDR), intermodulation distortion, or all of the above. Harmonic distortion is defined as the ratio of harmonics to fundamental when a (theoretically) pure sine wave is reconstructed, and is the most common specification. Spurious free dynamic range is the ratio of the worst spur (usually, but not necessarily always a harmonic of the fundamental) to the fundamental.

Code-dependent glitches will produce both out-of-band and in-band harmonics when the DAC is reconstructing a digitally generated sinewave (as in a Direct Digital Synthesis (DDS) system). The midscale glitch occurs twice during a single cycle of a reconstructed sinewave (at each midscale crossing), and will therefore produce a second harmonic of the sinewave, as shown in Figure 5.16. Note that the higher order harmonics of the sinewave, which alias back into the Nyquist bandwidth (DC to $f_S/2$), cannot be filtered.

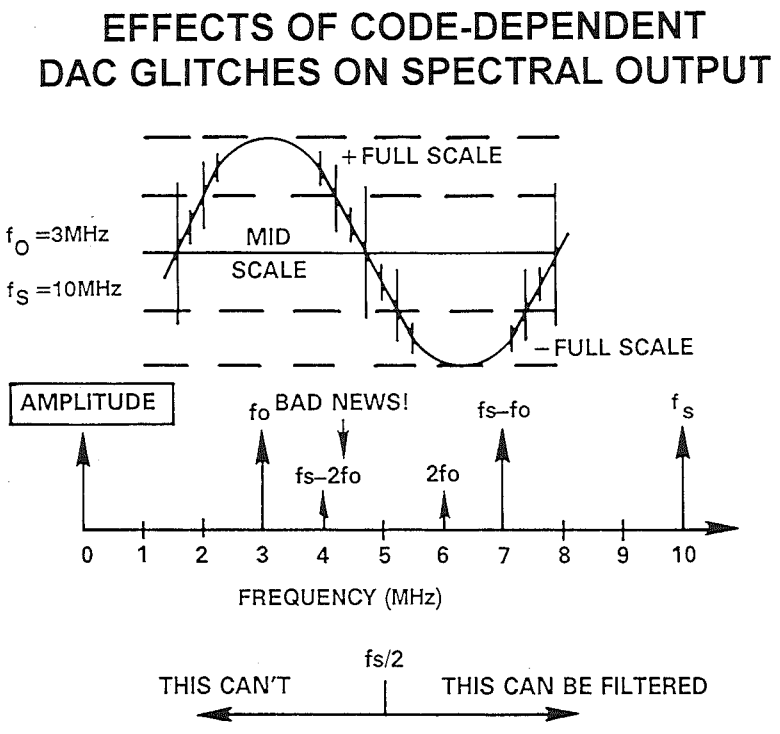


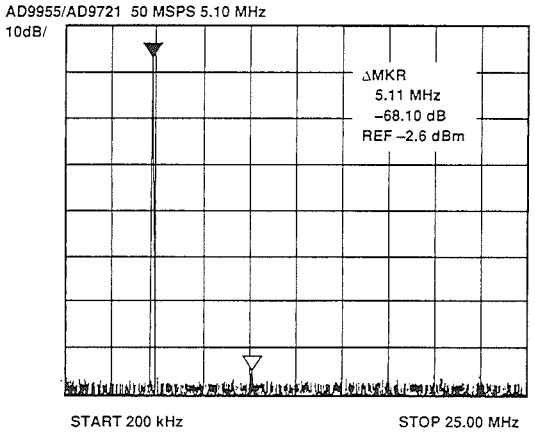
Figure 5.16

It is difficult to predict the harmonic distortion or SFDR from the glitch specification alone. Other factors, such as the overall linearity of the DAC, also contribute to distortion. It is therefore customary to test reconstruction DACs in the frequency domain (using a spectrum analyzer) at various clock rates and output frequencies. Typical spectral outputs for the AD9721 10 bit DAC are shown in Figure 5.17. The clock rate is 50MSPS, and the output frequencies are 5.10MHz and 16.60MHz, respectively. The AD9955 DDS chip is used to generate the digital sinewave. A detailed discussion of such systems is beyond the scope of this section; however a complete discussion of high speed DACs and DDS may be found in Section 17 of Reference 2.

DACs also produce spurious frequency domain products because of the quantization process (dividing the DAC output signal range into 2^N discrete values). Because quantization theory is identical for both ADCs and DACs, it is discussed in the next section.

AD9955 DDS AND AD9721 DAC SPECTRAL OUTPUT FOR CLOCK FREQUENCY OF 50MSPS

OUTPUT = 5.10MHz



OUTPUT = 16.60MHz

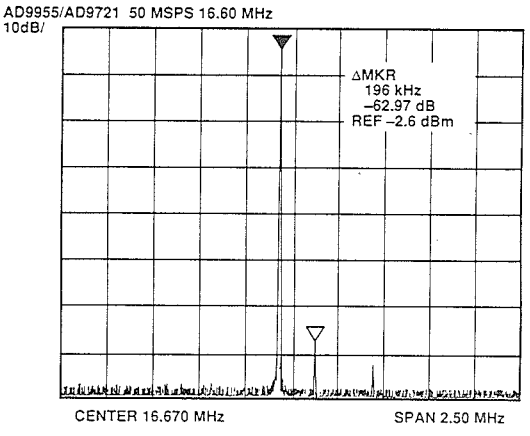


Figure 5.17

AC ERRORS IN ADCS

The AC specifications of ADCs include *harmonic* and *intermodulation distortion*, *Spurious Free Dynamic Range* (SFDR), *full-power bandwidth* (FPBW), *effective number of bits* (ENOB), and *sampling clock jitter*. We must also consider quantization noise.

When we discussed the DC performance of ADCs, we pointed out that the input of an ADC is analog and therefore has infinite resolution, but the output is quantized into 2^N steps. The difference between analog input and the exact value of the digital output it produces is the quantization error. The same phenomenon occurs in AC systems, producing quantization noise. Mathematically predicting the precise spectral content of the quantization noise as a function of the input signal is quite difficult. In many cases, however, the quantization noise is relatively uncorrelated to the input signal. When this is the case, the quantization noise behaves as broadband noise with a uniform spectral density between DC and one-half the sampling frequency.

The rms quantization noise voltage of an ADC in the Nyquist band (DC - $f_S/2$) is $q/\sqrt{12}$, where q is the value of the LSB. From this, we can calculate that the signal-to-noise ratio (SNR) of a perfect N-bit ADC with a full scale sine wave input signal is limited to (6.02N + 1.76) dB by quantization noise. Like the Johnson noise of a perfect resistor, quantization noise is a fundamental, and cannot be improved by changes in converter design. However, again like Johnson noise, we can redesign systems so that its effects are less damaging.

Quantization noise may be fundamental, but most high resolution ADCs have noise or distortion sources that are larger than the quantization noise, and which reduce their effective resolution to less than the number of bits given on their data sheets.

QUANTIZATION NOISE

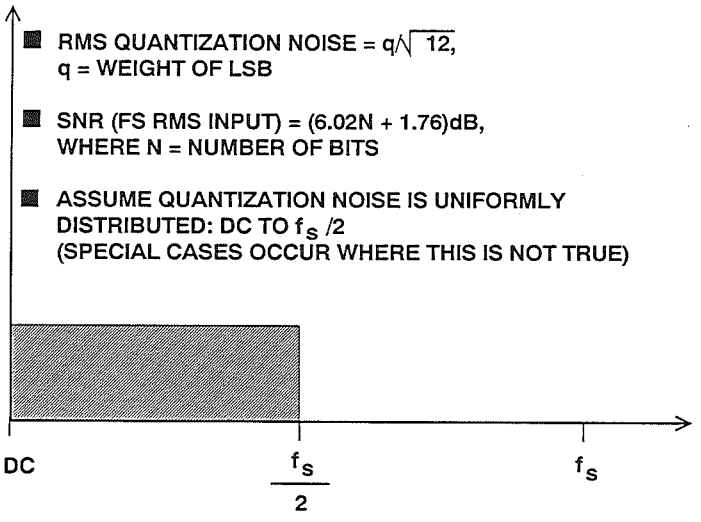


Figure 5.18

If we plot the gain of an amplifier with a small signal of a few millivolts or tens of millivolts, we find that as we increase the input frequency, there is a frequency at which the gain has dropped by 3 dB. This frequency is the upper limit of the "small signal bandwidth" of the amplifier and is set by the internal pole(s) in the amplifier response. If we drive the same amplifier with a large signal so that the output stage swings with its full rated peak to peak output voltage, we *may* find that the upper 3 dB point is at a lower frequency, being limited by the slew rate of the amplifier output stage. This high level 3 dB point defines the "large signal bandwidth" of the amplifier. When defining the large signal bandwidth of an amplifier, a number of variables must be considered, including the power supply, the output amplitude (if slew rate is the only limiting factor, it is obvious that if the signal amplitude is halved, the "large signal bandwidth" is doubled), and the load. Thus "large signal bandwidth" is a rather uncertain parameter in an amplifier, since it depends on so many uncontrolled variables - in cases where the large signal bandwidth is less than the small signal bandwidth, it is better to define the output slew rate and calculate the maximum output swing at any particular frequency. In an ADC, however, the maximum swing is always full-scale, and the load seen by the signal is defined. It is therefore quite reasonable to define the large signal bandwidth (or full-power bandwidth) of an ADC and report it on the data sheet.

ADC LARGE SIGNAL (OR FULL POWER) BANDWIDTH

- With Small Signal, the Bandwidth of a Circuit is limited by its Overall Frequency Response.
- At High Levels of Signal the Slew Rate of Some Stage May Control the Upper Frequency Limit.
- In Amplifiers There are so many Variables that *Large Signal Bandwidth* needs to be Redefined in every Individual Case, and *Slew Rate* is a more Useful Parameter for a Data Sheet.
- In ADCs the Maximum Signal Swing is the ADC's Full Scale Span, and is therefore Defined, so *Full Power Bandwidth* (FPBW) may Appear on the Data Sheet.
- HOWEVER the FPBW Specification Says Nothing About Distortion Levels. Effective Number of Bits (ENOB) is Much More Useful in Practical Applications.

Figure 5.19

However, the large signal bandwidth tells us the frequency at which the amplitude response of the ADC drops by 3 dB — it tells us nothing at all about the relationship between distortion and frequency. If we study the behavior of an ADC as its input frequency is increased, we discover that, in general, noise and distortion increase with frequency. This reduces the resolution that we can obtain from the ADC.

If we draw a graph of the ratio of signal to noise plus distortion of a converter against its input frequency, we find a much more discouraging graph than that of its frequency response. The ratio of signal to noise plus distortion can be expressed in dB or as an effective number of bits (ENOB). As we have seen above, the SNR of a perfect N-bit ADC (with a full-scale sine input) is (6.02N + 1.76) dB. If we have an ADC with an SNR of X dB with a particular input, then we can solve the equation:

$$\text{ENOB} = \frac{\text{XdB} - 1.76\text{dB}}{6.02\text{dB}}$$

and determine the *effective* resolution of the converter with that input. A graph of ENOB against variations of input amplitude and frequency can be depressing when we see just how little of the DC resolution of the ADC can actually be used, but can sometimes show interesting features: the ADC in Figure 5.21, for instance, has a larger ENOB for signals at 10% of FS at 1 MHz than for FS signals of the same frequency - a simple frequency response curve cannot have plots crossing in this way.

EFFECTIVE NUMBER OF BITS (ENOB) INDICATES DYNAMIC PERFORMANCE OF ADCs

- SNR = 6.02N + 1.76dB (THEORETICAL)
- ADC ACHIEVES SNR = XdB (ACTUAL)
- $\text{ENOB} = \dfrac{\text{XdB} - 1.76\text{dB}}{6.02\text{dB}}$
- ENOB INCLUDES: NOISE AND DISTORTION, DC TO $f_S/2$

Figure 5.20

ADC GAIN AND ENOB VERSUS FREQUENCY SHOWS IMPORTANCE OF ENOB SPECIFICATION

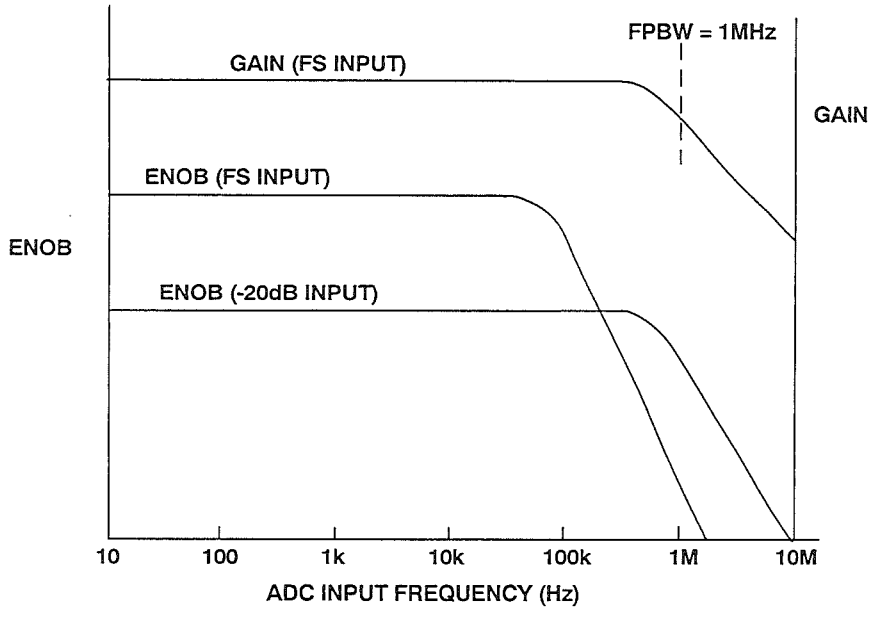


Figure 5.21

The causes of the loss of ENOB at higher input frequencies are varied. The linearity of the ADC transfer function probably degrades as the input frequency increases, thereby causing higher levels of distortion. Another reason that the SNR of an ADC decreases with input frequency may be deduced from Figure 5.22, which shows the effects of phase jitter on the sampling clock of an ADC. The phase jitter causes a voltage error which is a function of slew rate and results in an overall degradation in SNR as shown in Figure 5.23. This is quite serious, especially at higher input/output frequencies. Therefore, extreme care must be taken to minimize phase noise in the sampling/reconstruction clock of any sampled data system. This care must extend to all aspects of the clock signal: the oscillator itself (for example, a 555 timer is absolutely inadequate, but even a quartz crystal oscillator can give problems if it uses an active device which shares a chip with noisy logic); the transmission path (these clocks are very vulnerable to interference of all sorts), and phase noise introduced in the ADC or DAC. A very common source of phase noise in converter circuitry is aperture jitter in the integral sample-and-hold (SHA) circuitry.

EFFECTS OF APERTURE AND SAMPLING CLOCK JITTER

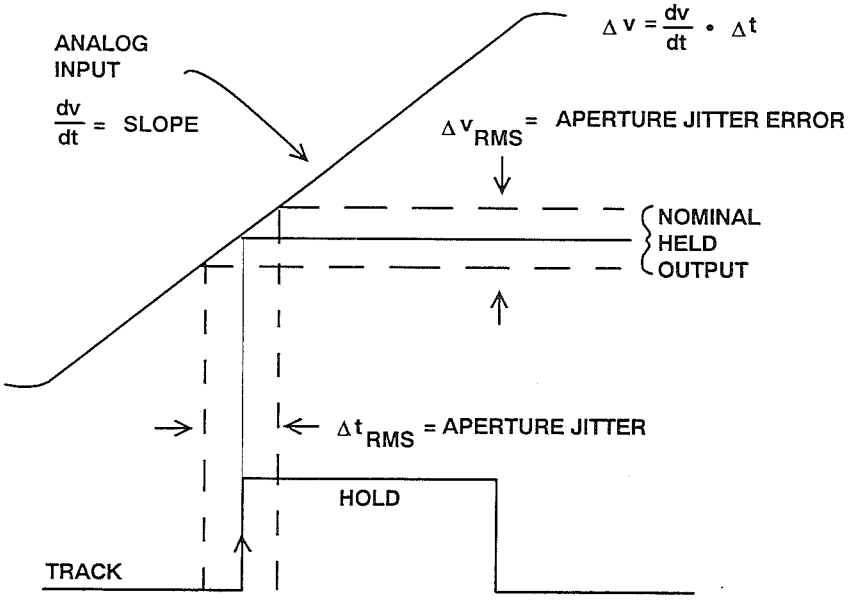


Figure 5.22

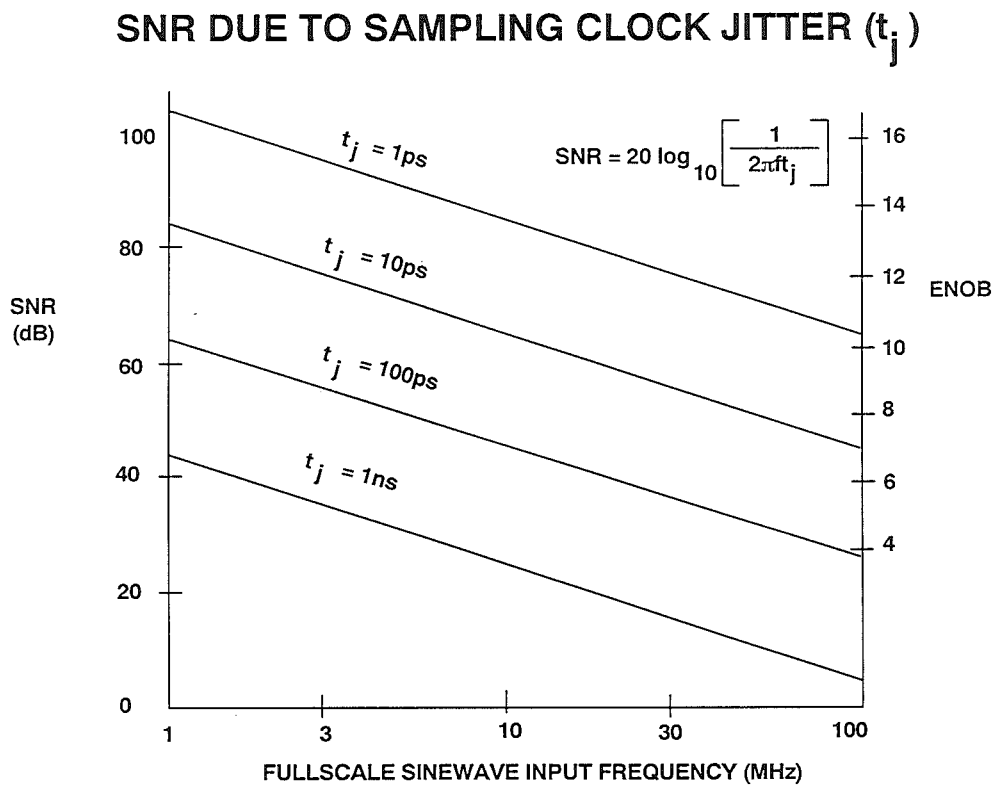


Figure 5.23

A decade or so ago sampling ADCs were built up from a separate SHA and ADC. Interface design was difficult, and a key parameter was aperture jitter in the SHA. Today, most sampled data systems use *sampling* ADCs which contain an integral SHA. The aperture jitter of the SHA may not be specified as such, but this is not a cause of concern if the SNR or ENOB is clearly specified, since a guarantee of a specific SNR is an implicit guarantee of an adequate aperture jitter specification. However, the use of an additional high-performance SHA will sometimes improve the high-frequency ENOB of a sampling ADC, and may be more cost-effective than replacing the ADC with a more expensive one.

It should be noted that there is also a fixed component which makes up the ADC aperture time. This component, usually called *effective aperture delay time*, does not produce an error. It simply results in a time offset between the time the ADC is asked to sample and when the actual sample takes place (see Figure 5.24). The variation or tolerance placed on this parameter from part to part is important in simultaneous sampling applications or other applications such as I and Q demodulation where several ADCs are required to track each other.

EFFECTIVE APERTURE DELAY TIME

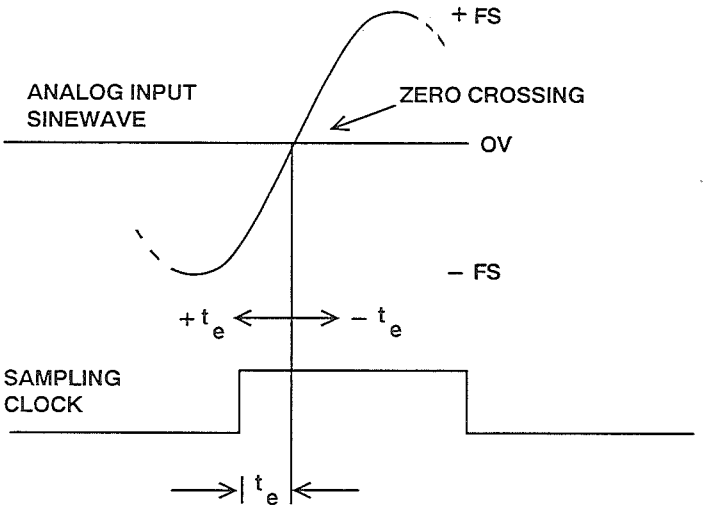


Figure 5.24

The distortion produced by an ADC or DAC cannot be analyzed in terms of second and third-order intercepts, as in the case of an amplifier. This is because there are two components of distortion in a high performance data converter. One component is due to the non-linearity associated with the analog circuits within the converter. This non-linearity has the familiar "bow" or "s"-shaped curve shown in Figure 5.23. (It may be polynomial or logarithmic in form). The distortion associated with this type of non-linearity is sometimes referred to as *soft* distortion and produces low-order distortion products. This component of distortion behaves in the traditional manner, and is a function of signal level. In a practical data converter, however, the soft distortion is usually much less than the other component of distortion, which is due to the differential nonlinearity of the transfer function. The converter transfer function is more likely to have discrete points of discontinuity across the signal range as shown in Figure 5.25.

TRANSFER CHARACTERISTICS FOR "SOFT" AND "HARD" DISTORTION IN DATA CONVERTERS

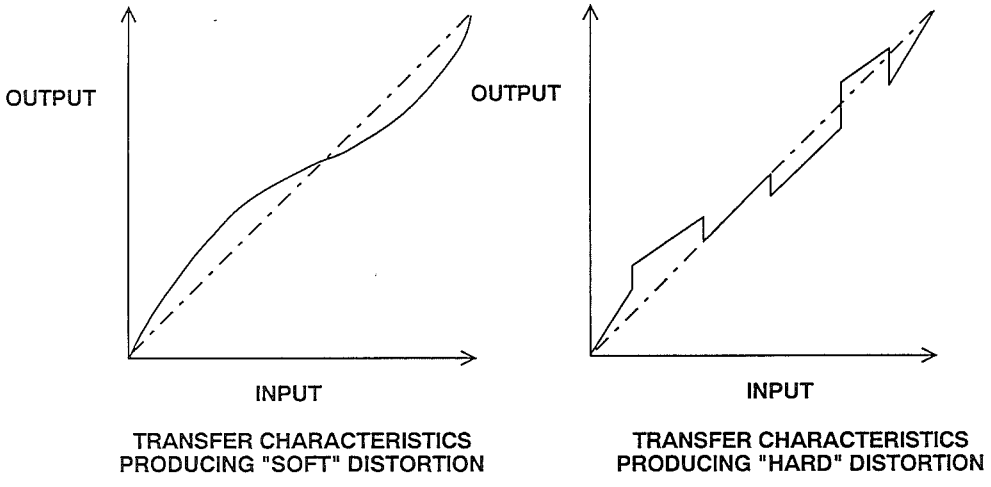


Figure 5.25

The actual location of the points of discontinuity depends on the particular data converter architecture, but nevertheless, such discontinuities occur in practically all converters. Non-linearity of this type produces high-order distortion products which are relatively unpredictable with respect to input signal level, and therefore such specifications as *third order intercept point* may be less relevant to converters than to amplifiers and mixers. For lower-amplitude signals, this constant level *hard* distortion causes the SFDR of the converter to *decrease* as input amplitude decreases. The soft distortion in a well-designed converter is only significant for high frequency large-amplitude signals where it may rise above the hard distortion floor.

COMPONENTS AND PROCESSES FOR DATA CONVERTERS

Data converters may be constructed with a wide range of technologies, although the majority are monolithic integrated circuits (ICs) of various types. There are many IC technologies, and many tradeoffs between them. In general, data converters require logic circuitry, switches, and some precision analog circuitry.

COMPONENTS FOR DATA CONVERTERS

- Data Converter Component Requirements:
  - Good Logic, Good Switches
  - Good Analog Circuitry:

    Amplifiers, Comparators,
    References, Resistors

Figure 5.26

This precision analog circuitry may include stable, accurate voltage or current references, stable precision-matched resistors, and precision amplifiers and comparators. The processes used to manufacture good amplifiers, comparators and references do not, in general, make good logic, so manufacturers of early data converters used multi-chip hybrids to overcome these process limitations. Hybrid technology tends to be expensive, so converter manufacturers have been in the forefront of the development of IC processes capable of both analog and digital performance, so that data converters can be manufactured on a single chip. In general, they have been successful, but even today, some of the highest performance converters require chips from more than one process.

Since the name *hybrid* is perceived as implying *expensive*, some of these converters have had their process renamed. Where two monolithic chips from different technologies share a package, but do not require a ceramic substrate or any other integrated components, the structure, which is much less expensive than a classical hybrid, is sometimes known as *compound monolithic* rather than *hybrid*.

HYBRID CONVERTERS

- **Early Data Converters used hybrid technology to achieve performance unavailable from any single monolithic technology.**
- **Even today, some of the best converters cannot use any available monolithic technology and are hybrid.**
- ***Compound Monolithic* is a term for a simpler (and cheaper) hybrid technology where two monolithic chips from different technologies are mounted together in a single package, but without a ceramic substrate or other components.**

Figure 5.27

In general, bipolar processes make good, stable, low-noise amplifiers, comparators and references. Bipolar logic, however, though fast, occupies large chip areas and is therefore expensive. Also, processes which make good bipolar logic tend to be noisier than analog processes. CMOS is capable of making high-density, low-power logic very efficiently, but although it can also make good analog switches, it is not very suitable for amplifiers, and almost incapable of making references.

These considerations caused process technologists to combine bipolar and CMOS processes to achieve both low-power, high density logic and high-accuracy, low-noise analog circuitry on a single chip. The resulting processes are more complex, and therefore more expensive, than simple bipolar and CMOS processes, but do have better mixed performance. They include BIMOS processes which are basically bipolar processes to which CMOS structures have been added, and linear compatible CMOS (LC^{2}MOS or LCCMOS) which is basically CMOS with added bipolar capability. However, the compromises necessary to combine features mean that neither BIMOS nor LCCMOS offers quite as good performance as its parent process does in *its* specialty. Thus, BIMOS and LCCMOS have not replaced bipolar or CMOS technology, but designers now have four processes from which to choose when designing a data converter.

Of course, to suggest that there is only *one* of each of these process is a gross simplification, and omits all considerations of such process developments as complimentary bipolar (CB) and dielectrically isolated (DI) processes. Since this section is concerned with converter structures, however, it is not really appropriate to discuss IC processes in detail. There is, however, one process technology which does deserve special mention, since it is crucial to the manufacture of any data converter requiring stable precision resistors. This is thin-film resistor technology.

MONOLITHIC CONVERTER PROCESSES

- ***Bipolar*** **processes have good analog performance but less good logic and switches.**
- ***CMOS*** **processes make excellent logic and switches but relatively poor amplifiers and references.**
- **Processes combining the two (*BiMOS*, *LCCMOS*, etc.) tend to be more complex and expensive and have slightly less performance than the sum of the two but are very convenient.**
- **Good designers choose the best process for the circuit to be designed.**

Figure 5.28

Analog Devices has spent many man-years of effort developing the ability to deposit stable thin-film resistors on integrated circuit chips, and yet more man-years in learning to laser trim them. All the processes mentioned above can incorporate such resistors. They have stability of <20 ppm/°C, and matching to within 0.005%. Actually, the resistors can be made to match to within 0.01% or better without laser trimming, but to achieve this they must be very large. In practice, if resistors must match to better than 0.1 or 0.05%, it is more economical to laser trim them than to design them to meet the specification without trimming. Most of the resistors in the next sections are these laser-trimmed SiCr thin-film resistors.

THIN FILM RESISTORS

- One of the key technologies for making many types of monolithic data converters is the ability to deposit accurate, stable SiCr resistors on monolithic chips.
- Some converters use these resistors as fabricated, others require the additional accuracy and economy of laser trimming.
- Parameters include:
  - Matching to 0.005%
  - Temperature Coefficient < 20ppm/°C
  - Differential TC < 0.2ppm/°C
  - Long Term Stability ≤1ppm/1000 hours (drunkard's walk)

Figure 5.29

DAC STRUCTURES

It is reasonable to consider a changeover switch, switching an output between a reference and ground, or between equal positive and negative reference voltages, as a 1-bit DAC (see Figure 5.30). Such a simple device is a component of many more complex DAC structures and is used, with oversampling, as the basic analog component in the sigma-delta DAC we shall discuss later. Nevertheless, it is a little too simple to require discussion, and it is more rewarding to consider more complex structures.

The DACs most commonly used as examples of simple DAC structures are binary weighted DACs or ladder networks, but these, though simple in structure, require quite complex analysis. The simplest structure of all, after the changeover switch mentioned above, is the Kelvin divider shown in Figure 5.31. An N-bit version of this DAC simply consists of 2^N equal resistors in series. The output is taken from the appropriate tap by closing one of the 2^N switches (there is some slight digital complexity involved in decoding to 1 of 2^N switches from N-bit data, but digital circuitry is comparatively cheap).

A CHANGEOVER SWITCH IS A 1-BIT DAC

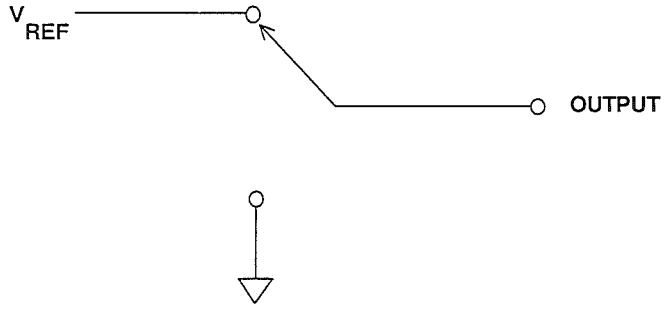


Figure 5.30

SIMPLEST MULTI-BIT DAC: THE KELVIN DIVIDER

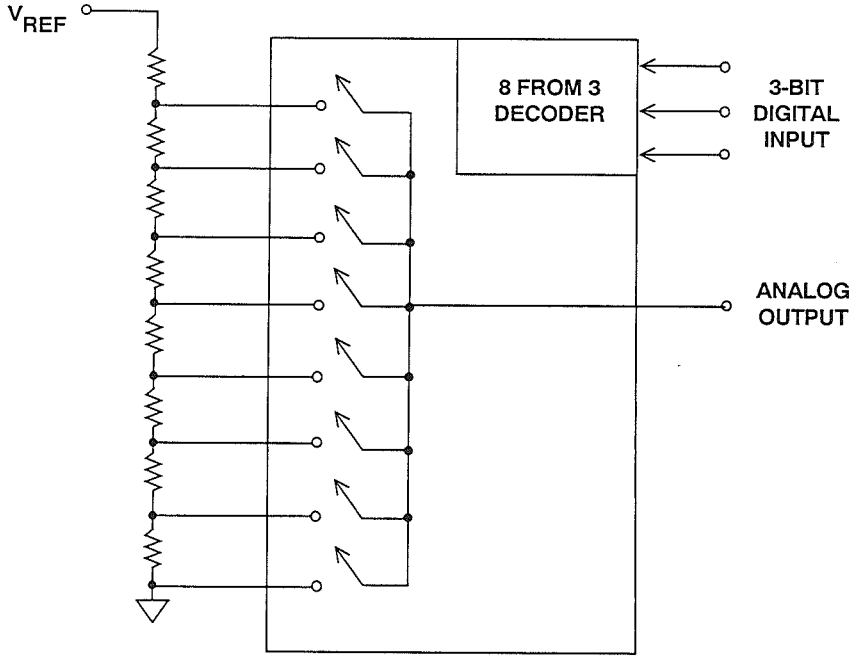


Figure 5.31

This architecture is simple, has a voltage output (but a code-varying Z_{OUT}), and is inherently monotonic (even if a resistor is zero, $OUTPUT_N$ cannot exceed $OUTPUT_{N+1}$). It is linear if all the resistors are equal, but may be made deliberately non-linear if a non-linear DAC is required. Since only two switches operate during a transition, it is a low-glitch architecture. Its major drawback is the large number of resistors required for high resolution, and as a result it is not commonly used (the only commercially available DAC of this type is a fast, low-glitch 8-bit device from Philips) — but, as we shall see later, it is used as a component in more complex DAC structures.

There is an analogous current output DAC which consists, again, of 2^N resistors (or current sources), but in this case they are all connected in parallel between the reference voltage input and the virtual ground output (see Figure 5.32).

THE SIMPLEST CURRENT OUTPUT DAC

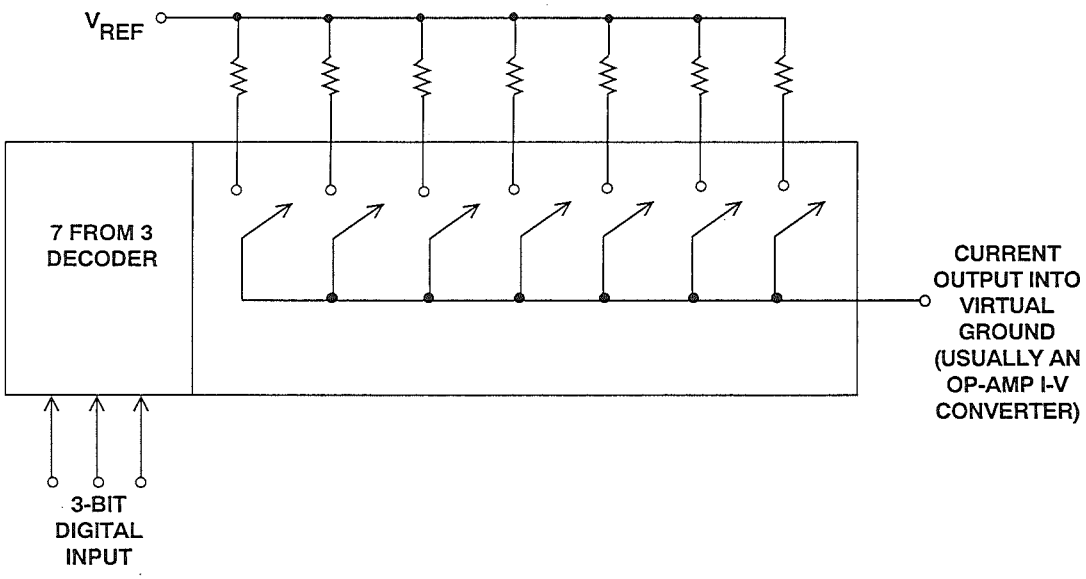


Figure 5.32

In this DAC, once a resistor is switched into circuit by increasing digital code, any further increases do not switch it out again. The structure is thus inherently monotonic, irrespective of inaccuracies in the resistors, and like the previous case, may be made intentionally non-linear where a specific non-linearity is required. Again, as in the previous case, the architecture is rarely, if ever, used to fabricate a complete DAC because of the large numbers of resistors and switches required. However, it is often used as a component in a more complex DAC structure.

Unlike the Kelvin divider, this type of DAC does not have a unique name, although both types are referred to as "fully decoded DACs" or "thermometer DACs".

Fully-decoded DACs are often used as components of more complex DACs. The most common are "segmented DACs" where part of the output of a fully decoded DAC is further subdivided. The structure is used because the fully decoded DAC is inherently monotonic, so if the subdivision is also monotonic, the whole resulting DAC is also monotonic.

A voltage segmented DAC (see Figure 5.33) works by further sub-dividing the voltage across one resistor of a Kelvin divider. The sub-division may be done with a further Kelvin divider (in which case the whole structure is known as a "Kelvin-Varley divider", or with some other DAC structure. Commercial 16-bit DACs exist with both arrangements: the AD569 uses two cascaded 8-bit Kelvin dividers in a Kelvin-Varley structure, a total of 512 resistors, to give a 16-bit monotonic but only 13-bit linear DAC (this linearity problem is one of resistor size - with 512 resistors on a chip, they are too small to laser trim, let alone to be intrinsically matched to 16-bits), and the AD7846 uses a 12-bit ladder network in the voltage mode connected across one resistor of a 4-bit Kelvin divider.

In a true current-segmented DAC using fully-decoded MSBs (see Figure 5.34), each current source has a three-way switch and is switched off, or to a further current steering DAC, or directly to the output. This architecture, too, is inherently monotonic.

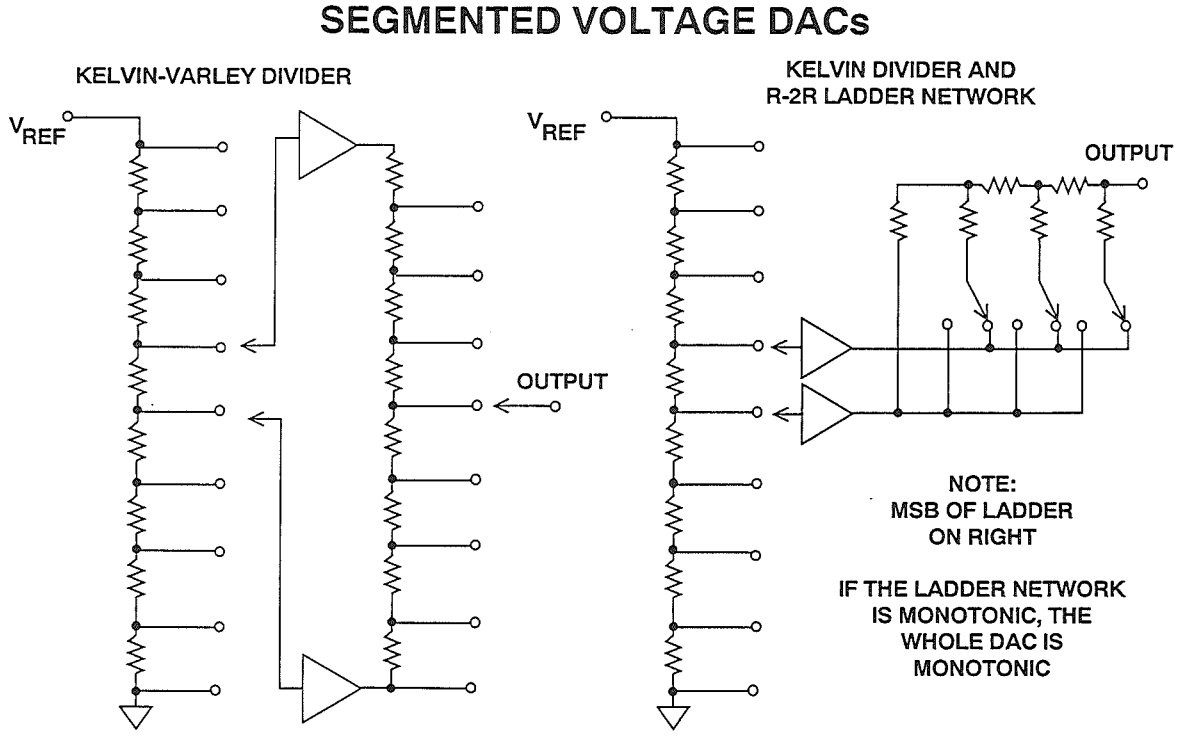


Figure 5.33

SEGMENTED 4-BIT CURRENT OUTPUT DAC

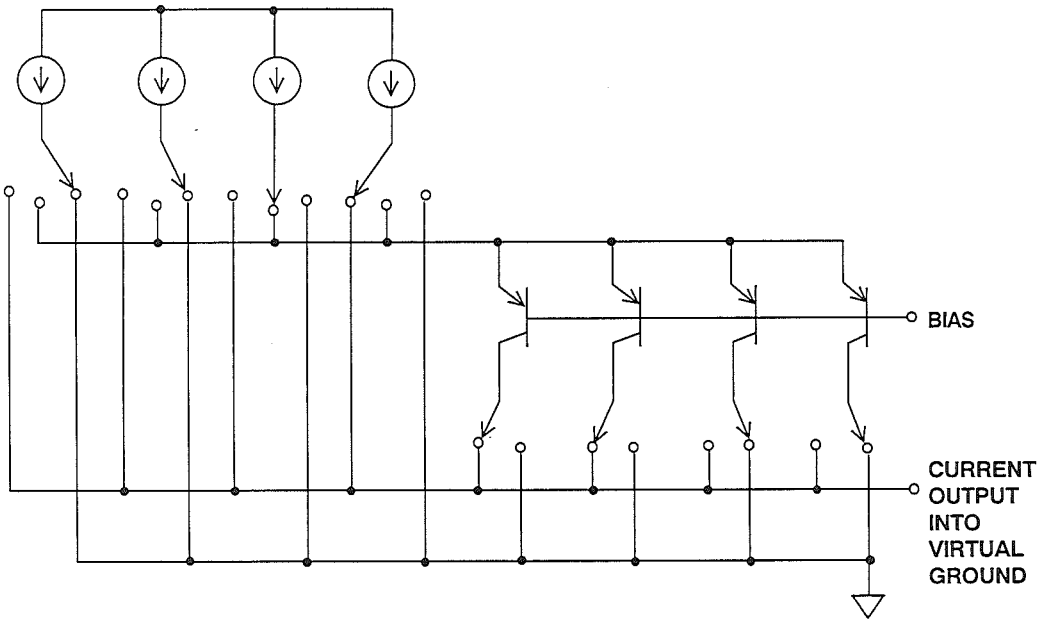
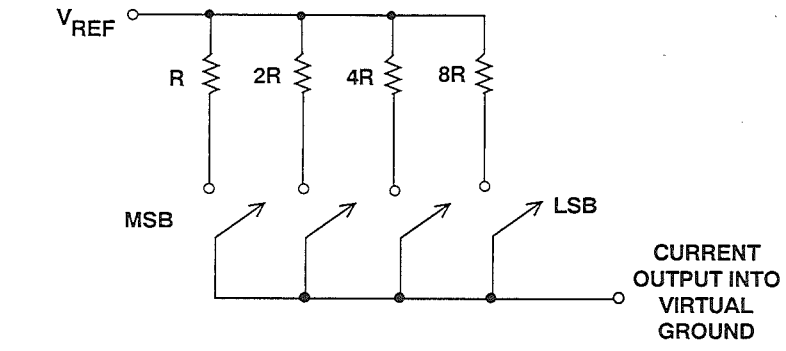


Figure 5.34

The simplest textbook example of a DAC, the binary-weighted DAC shown in Figure 5.35 is not inherently monotonic and is very hard to manufacture successfully. An N-bit DAC of this type consists of N resistors in the ratio $1:2:4:8:\ldots:2^{N-1}$. The LSB switches the 2^{N-1} resistor, the MSB the 1 resistor, etc. The theory is simple, but the practical problems of manufacturing an IC of an economical size with resistor ratios of even 128:1 for an 8-bit DAC are enormous, especially as the resistors must have matched temperature coefficients.

BINARY WEIGHTED CURRENT OUTPUT DAC



■ DIFFICULT TO FABRICATE IN IC FORM DUE TO LARGE RESISTOR RATIO

Figure 5.35

DAC USING CASCADED BINARY QUADS

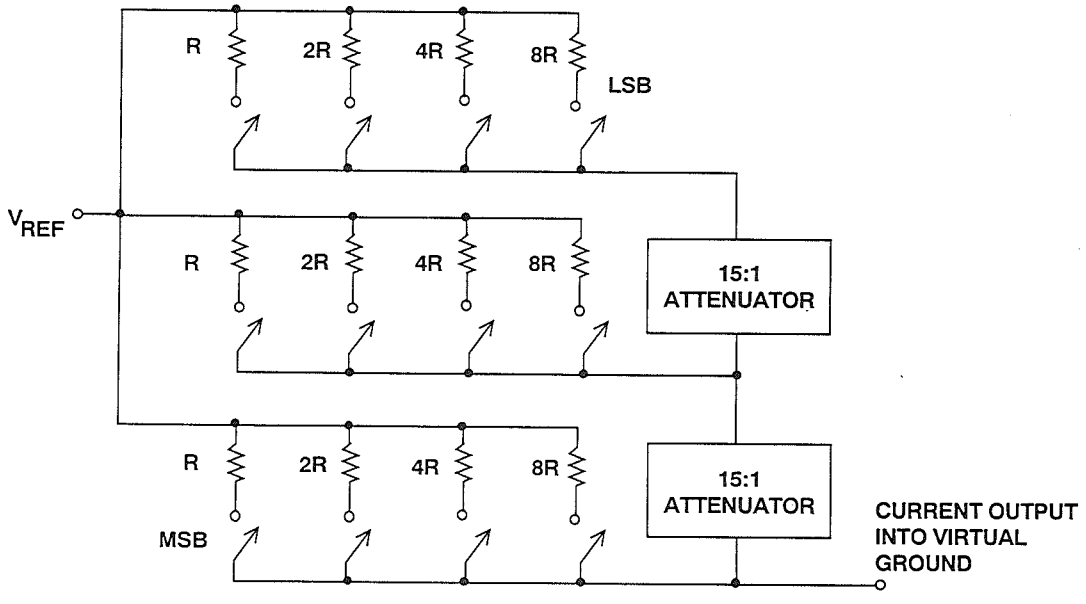


Figure 5.36

If the MSB resistor is even a little high in value, the MSB current will be less than the sum of all the other bit currents, and the DAC will not be monotonic (the differential non-linearity of most types of DAC is worst at major bit transitions). This architecture is never used in integrated circuit DACs, although, again, it has been used as a component of a more complex structure.

Figure 5.36 shows a 12-bit DAC made from three 4-bit binary weighted DACs with attenuators. In the early days of IC DACs, this architecture was sometimes used, but it was less than ideal, requiring 15:1 resistor ratios. The ideal resistor ratio for precision is as close as possible to 1:1, and the R:2R ladder network is not too far from this.

The R:2R ladder is shown in Figure 5.37 and uses resistors of only two different values: their ratio is 2:1. An N-bit DAC requires 2^N-1 resistors. There are two ways in which the R:2R ladder network may be used as a DAC, known respectively as the voltage mode, and the current mode. Each has its advantages and disadvantages.

4-BIT R-2R LADDER NETWORK

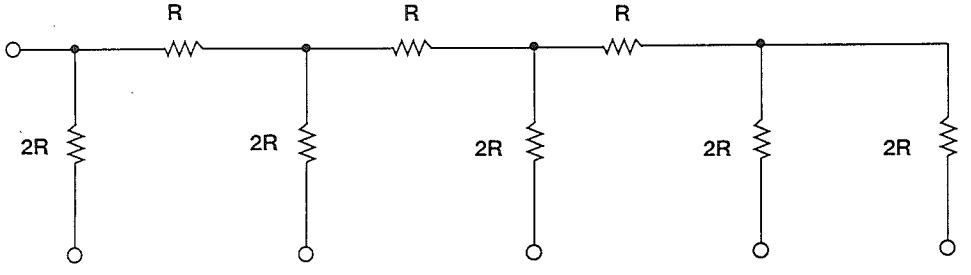


Figure 5.37

In the voltage mode, the "rungs" or arms of the ladder are switched between V_{ref} and ground, and the output is taken from the end of the ladder, as shown in Figure 5.38. The output may be taken as a voltage, but the output impedance is independent of code, so it may equally well be taken as a current into a virtual ground.

The voltage output is an advantage of this mode, as is the constant output impedance, which eases the stabilization of any amplifier on the output node. Additionally, the switches switch the arms of the ladder between a low impedance V_{ref} connection and ground, so capacitive glitch currents tend not to flow in the load. On the other hand, the switches must operate over a wide voltage range (V_{ref} to ground), which is difficult from a design and manufacturing viewpoint, and the reference input impedance varies widely with code, so that the reference input must be driven from a very low impedance, and the gain of the DAC cannot be adjusted with a preset resistor in series with the V_{ref} terminal.

VOLTAGE-MODE LADDER NETWORK DAC

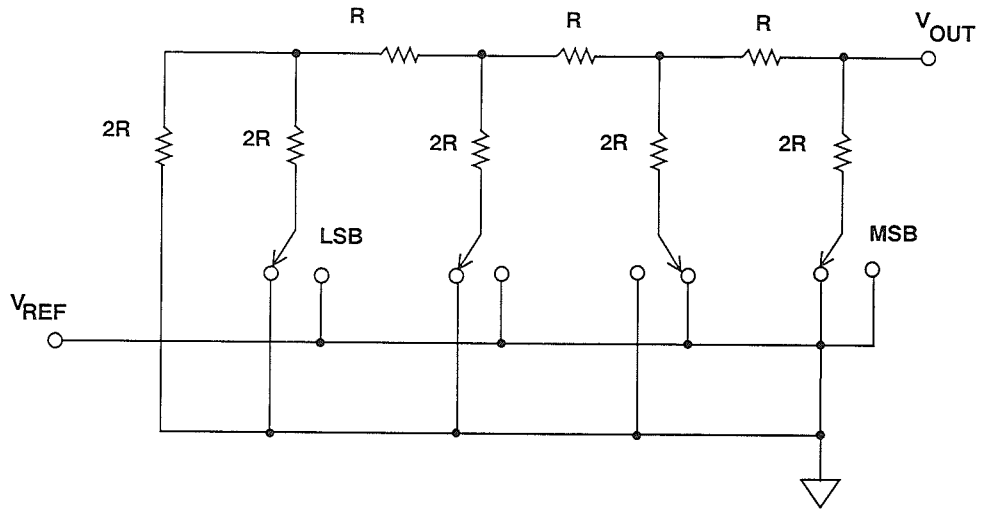


Figure 5.38

The gain of a DAC in the current mode configuration shown in Figure 5.39 may be adjusted with a series resistor since, in the current mode, the end of the ladder, with its code-independent impedance, is used as the V_{ref} terminal, and the ends of the arms are switched between ground and an output line (which must be held at ground potential). The normal connection of a current mode ladder network output is to an op-amp configured as an I-V converter, but stabilization of this op-amp may be complicated by the DAC output impedance variation with code.

CURRENT-MODE LADDER NETWORK DAC

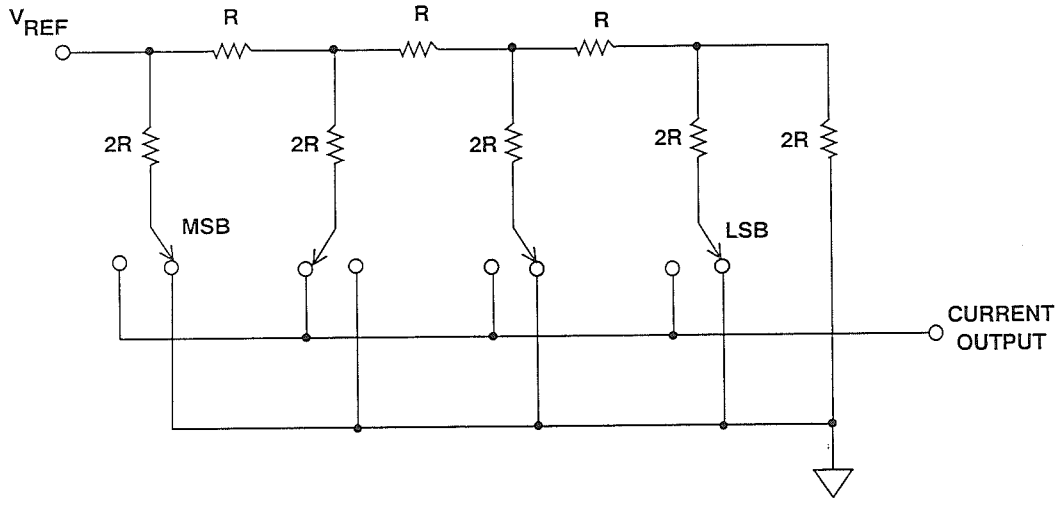


Figure 5.39

Current mode operation has higher glitch, though, since the switches connect directly to the output line, but as the switches of a current-mode ladder network are always at ground potential, their design is less demanding and, in particular, their voltage rating does not affect the reference voltage rating. If switches capable of carrying current in either direction (such as MOS devices) are used, the reference voltage may have either polarity, or may even be AC. Such a structure is one of the most common types of multiplying DAC (MDAC).

In all DACs, the output is the product of the reference voltage and the digital code, so in that sense, all DACs are multiplying DACs, but many DACs operate well only over a limited range of V_{ref}. True MDACs, however, are designed to operate over a wide range of V_{ref}. A strict definition of a multiplying DAC demands that its reference voltage range includes 0V, and many, especially current mode ladder networks with CMOS switches, permit positive, negative, and AC V_{ref}. DACs which do not work down to 0V V_{ref} are still useful, however, and types where V_{ref} can vary by 10:1 or so are often called MDACs, although a more accurate description might be "semi-multiplying" DACs.

MULTIPLYING DACs (MDACs)

- In all DACs the output is the product of the reference voltage and the digital code.
- Most DACs work over a limited range of reference voltages.
- DACs which work well with reference voltages which include zero volts are known as *Multiplying DACs.*
- Many MDACs work with bipolar and AC reference voltages.
- DACs which work with a large range of reference voltages, but not down to zero, are not true MDACs but are sometimes called MDACs. It is better to use the term *semi-multiplying DACs.*

Figure 5.40

The ladder network is certainly the most common DAC architecture. Where higher resolutions are required, however, trimming a ladder network can be demanding. A common way of minimizing the problem is to build a current mode DAC with 2 or 3 fully decoded MSBs, and a ladder network for the remainder of the structure as shown in Figure 5.41.

A SEGMENTED LADDER DAC

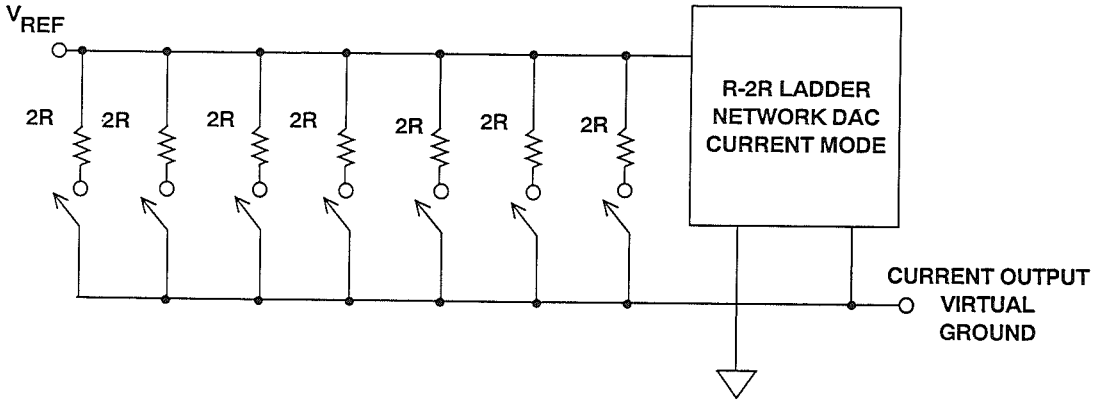


Figure 5.41

This arrangement is sometimes referred to as a *segmented* DAC, but the description is not strictly accurate, as even if the ladder network is monotonic, it is not inherently certain that the overall DAC will be, although it is comparatively easy to laser trim such an arrangement to monotonicity.

Digital audio requires DACs with resolutions of over 16-bits, good linearity, and low cost. One way to make them is to use a ladder with several decoded MSBs as described above, but this is likely to have comparatively large DNL at the MSB transition, which is just where low DNL is needed for low-level audio distortion. This problem can be avoided by using a digital adder to put a digital offset in the DAC code, so that the MSB transition of the input code is well offset from the mid-point of the DAC transfer characteristic, and then using an analog offset on the DAC output to restore the DC level at the crossover (see Figure 5.42).

20-BIT AUDIO DAC WITH OFFSET MSB TRANSITION

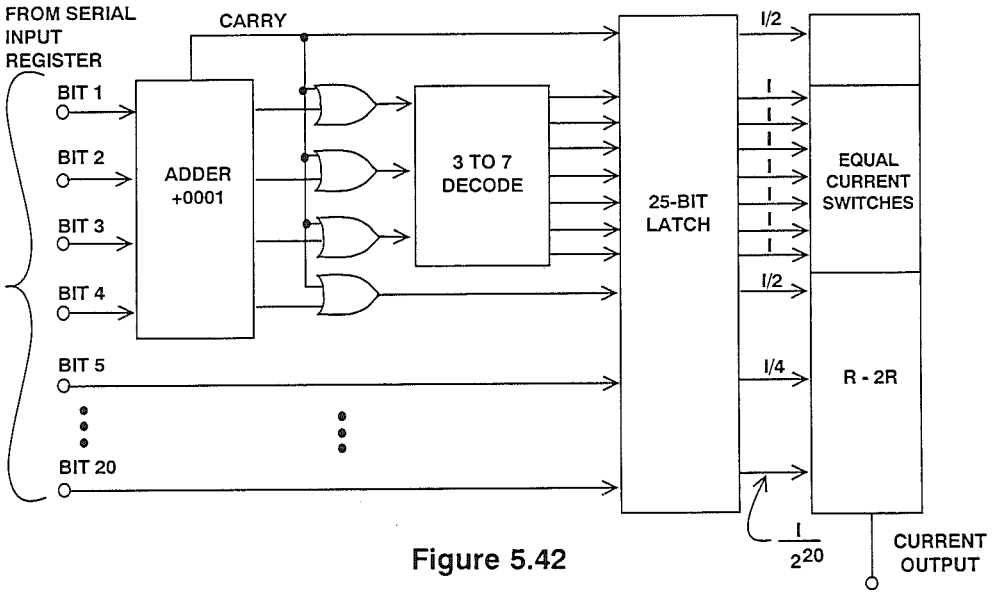


Figure 5.42

Another way of obtaining high resolution is to use oversampling techniques and a 1-bit DAC. The technique, known as sigma-delta (Σ–Δ), is computation intensive, so has only recently become practical for the manufacture of high resolution DACs, but since it uses a 1-bit DAC, it is intrinsically linear and monotonic.

A Σ–Δ DAC, unlike the Σ–Δ ADC we shall discuss later, is entirely digital (see Figure 5.43). It consists of an "interpolation filter" (a digital circuit which accepts data at a low rate, inserts zeros at a high rate, and then applies a digital filter algorithm and outputs data at high rate), a Σ–Δ modulator (which effectively acts as a low pass filter to the signal but as a high pass filter to the quantization noise, and converts the resulting data to a high speed bit stream), and a 1-bit DAC (which, as we mentioned earlier, is simply a changeover switch) connected to equal positive and negative reference voltages. The output is filtered in an external analog LPF.

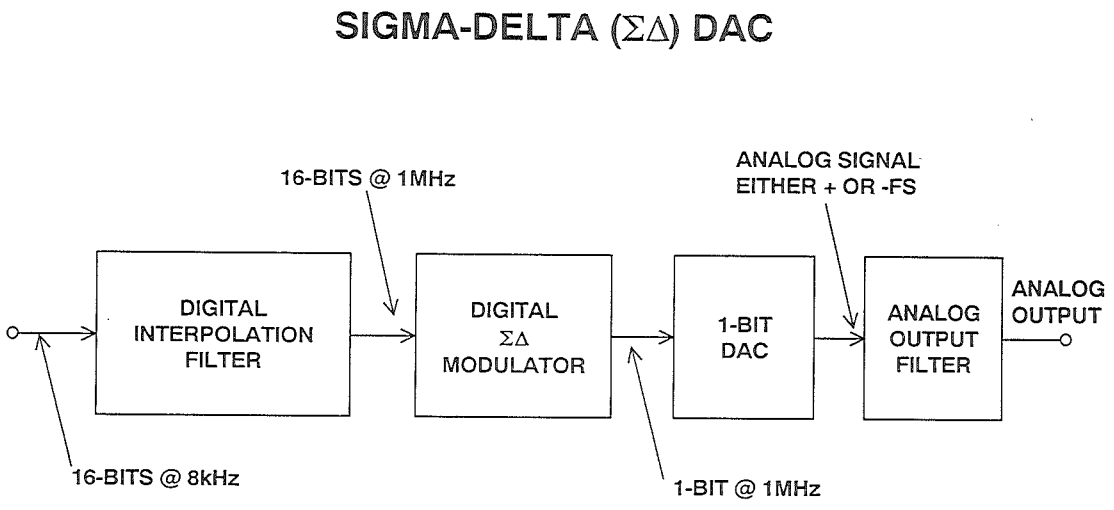


Figure 5.43

In theory, a Σ–Δ DAC is perfectly linear and has very low noise. In practice, they are very good, but, because of the difficulty of having an adequately fast clock with low enough phase noise and still separating digital noise from the analog signal, are still (as of mid 1994) marginally less good for the highest quality digital audio than the best traditional DACs. Nevertheless, Σ–Δ DACs are the optimum choice for a large number of low price, medium to high quality reconstruction DAC applications.

DAC LOGIC

The earliest monolithic DACs contained little, if any, logic circuitry, and parallel data had to be maintained on the digital input to maintain the digital output. Today almost all DACs are latched, and data need only be written once, not maintained.

There are innumerable variations of DAC input structure which will not be discussed here, but the majority today are "double-buffered". A double-buffered DAC has two sets of latches. Data is initially latched in the first rank and subsequently transferred to the second as shown in Figure 5.44. There are two reasons why this arrangement is useful.

The first is that it allows data to enter the DAC in many different ways. A DAC without a latch, or with a single latch, must be loaded with all bits at once, in parallel, since otherwise its output during loading may be totally different from what it was or what it is to become. A double-buffered DAC, on the other hand, may be loaded with parallel data, or with serial data, or with 4-bit or 8-bit words, or whatever, and the output will be unaffected until the new data is completely loaded and the DAC receives its update instruction.

DOUBLE-BUFFERED DAC PERMITS COMPLEX INPUT STRUCTURES AND SIMULTANEOUS UPDATE

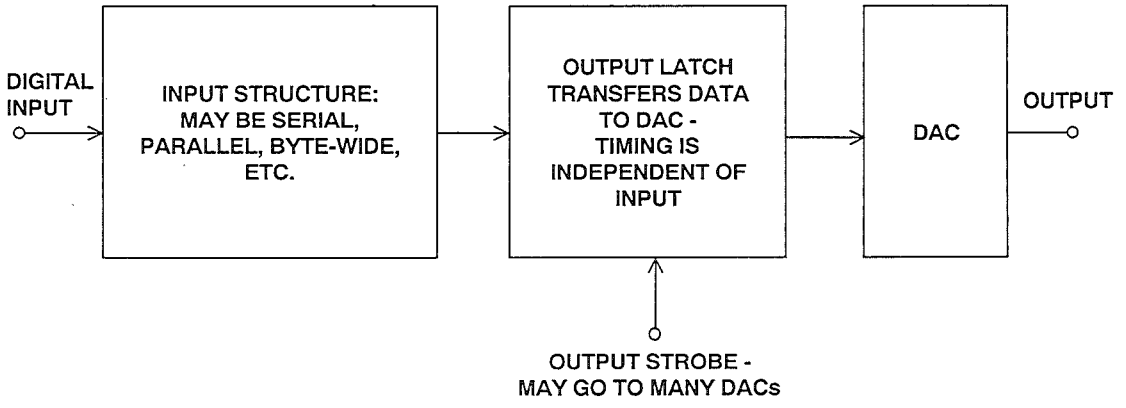


Figure 5.44

The other convenience of the double-buffered structure is that many DACs may be updated simultaneously: data is loaded into the first rank of each DAC in turn, and when all is ready, the output buffers of all DACs are updated at once. There are many DAC applications where the output of several DACs must change simultaneously, and the double-buffered structure allows this to be done very easily.

Most early monolithic high resolution DACs had parallel or byte-wide data ports and tended to be connected to parallel data buses and address decoders and addressed by microprocessors as if they were very small write-only memories (some DACs are not write-only, but can have their contents read as well - this is convenient for some applications but is not very common). A DAC connected to a data bus is vulnerable to capacitive coupling of logic noise from the bus to the analog output, and many DACs today have serial data structures. These are less vulnerable to such noise (since fewer noisy pins are involved), use fewer pins and therefore take less space, and are frequently more convenient for use with modern micro-processors, many of which have serial data ports. Some, but not all, of such serial DACs have data outputs as well as data inputs so that several DACs may be connected in series and data clocked to them all from a single serial port. The arrangement is referred to as "daisy-chaining".

SERIAL DACS

- If data is loaded serially into a DAC it requires fewer data pins.
- This saves space and also reduces capacitive noise coupling from data lines to the analog output.
- If the shift register of serial DAC has an output pin, a number of DACs may be connected in series (*daisy chained*) to a single serial data port.

Figure 5.45

Another development in DACs is the ability to make several on a single chip, which is useful to reduce PCB sizes and assembly costs. Today (mid-1994) it is possible to buy sixteen 8-bit, eight 12-bit, four 14-bit, or two 16-bit or 18-bit DACs in a single package. In the future, even higher densities are probable.

MULTIPLE DACS SAVE SPACE

- Several DACs on one chip are cheaper, smaller, and simpler to interface than several packages.
- Present limits (mid-1994) are:

16 × 8-bit

8 × 12-bit

2 × 16-bit or 18-bit

(Stereo Audio)

Figure 5.46

ADC STRUCTURES

In this section, we shall consider the structures and conversion algorithms of eight types of ADC, but, as in the DAC section, we shall not be concerned with many details of their digital interfaces, save to point out that if logic is cheap the results of an A-D conversion may easily be reformatted in any convenient way; parallel, serial or byte-wide. This means that it is generally possible to find a general purpose ADC with integral logic to provide an output data format well-suited to any particular application. Special purpose ADCs (high speed, high resolution, etc.) may only be available with a more limited range of options.

TYPES OF ANALOG-TO-DIGITAL CONVERTERS

- *Comparator*: A 1-bit ADC
- *Flash* : Fast, low-resolution, high power
- *Subranging*: Fast, high-resolution, complex
- *Integrating*: Slow, accurate, low power
- *Voltage-to-Frequency (VFC)*: High-resolution, low power, ideal for telemetry
- *Tracking*: Fast and slow, high-resolution
- *Successive Approximation*: Versatile, general purpose
- *Sigma-Delta*: Complex, low power, high-resolution

Figure 5.47

There are one or two practical points which are worth remembering about the logic of ADCs. On power-up, many ADCs do not have logic reset circuitry and may enter an anomalous logical state. One or two conversions may be necessary to restore their logic to proper operation so: (a) the first and second conversions after power-up should never be trusted, and (b) control outputs (EOC, DRDY, etc.) may behave in unexpected ways at this time (and not necessarily in the same way at each power-up), and (c) care should be taken to ensure that such anomalous behavior cannot cause system latch-up. For example, EOC (End Of Conversion) should not be used to initiate conversion if there is any possibility that EOC will not occur until the first conversion has taken place, as otherwise initiation will never occur.

Another detail which can cause trouble is the difference between EOC and DRDY (Data Ready). EOC indicates that conversion has finished, DRDY that data is available at the output. In some ADCs, data is not valid until several tens of nanoseconds *after* the EOC has become valid, and if EOC is used as a data strobe, the results will be unreliable.

As a final example, some ADCs use CS (Chip Select) edges to reset internal logic, and it may not be possible to perform another conversion without asserting or reasserting CS (or it may not be possible to read the same data twice, or both).

For more detail, it is important to read the whole data sheet before using an ADC since there are innumerable small logic variations from type to type. Unfortunately, many data sheets are not as clear as one might wish, so it is also important to understand the general principles of ADCs in order to interpret data sheets correctly. That is one of the purposes of this section.

BEWARE OF ADC LOGIC PITFALLS

- After power-up one or two conversions may be required before it runs right. The EOC (end of conversion) cannot always be trusted at this time.
- An ADC may not behave the same way every time it starts.
- EOC says conversion is finished.
  DRDY (data ready)says that data is valid.
  There may be tens of nanoseconds difference between the two.
- CS (Chip Select) may not just enable the data. It may reset things for the next conversion.
  In some converters, you *must* read the data!
  In some converters, you can't read the data twice!
  In some converters, you can't strap CS and forget it!
- READ THE DATA SHEET!!!

Figure 5.48

Comparators

As a changeover switch is a 1-bit DAC, so a comparator is a 1-bit ADC. If the input is above a threshold, the output has one logic value, below it has another. Moreover, there is no ADC architecture which does not use at least one comparator of some sort.

A COMPARATOR IS A 1-BIT ADC, AND ALL TYPES OF ADCs ARE BUILT WITH COMPARATORS

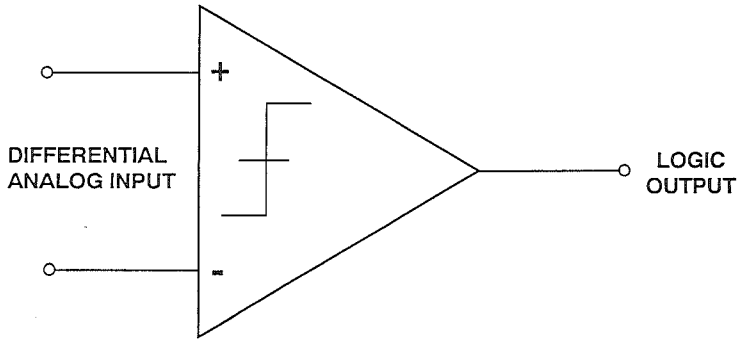


Figure 5.49

The most common comparator has some resemblance to an operational amplifier in that it uses a "long-tailed pair" of transistors or FETs as its input stage, but unlike an op amp, it does not use external feedback, and its output is a logic level indicating which of the two inputs is at the higher potential. Some comparators have a millivolt or two of hysteresis to encourage "snap" action and to prevent local feedback from causing instability in the transition region.

It is not usual to use a discrete comparator as an element in a more complex ADC so this paper will not discuss the structure or use of comparators in detail (but more information may be found in Reference 3 and Section 8 of this book). Where comparators are incorporated into IC ADCs, their design must consider resolution, speed, overload recovery, power dissipation, offset voltage, bias current, and the chip area occupied by the architecture which is chosen.

Flash or Parallel ADCs

Flash ADCs (sometimes called *parallel* ADCs) are the fastest type of ADC and use large numbers of comparators. An N-bit flash ADC consists of 2^N resistors and 2^N-1 comparators arranged as in Figure 5.50. It is clear that each comparator has a reference voltage which is 1 LSB higher than that of the one below it in the chain. For a given input voltage, all the comparators below a certain point will have their input larger than their reference and a "1" logic output, and all the comparators above that point will have a reference larger than input and a "0" logic output. The 2^N-1 comparator outputs therefore behave in a way analogous to a mercury thermometer, and the output code at this point is sometimes called a *thermometer* code. Since 2^N-1 data outputs are not really practical, they are processed by a priority encoder to an N-bit output.

The signal is applied to all the comparators at once, so the thermometer output is delayed by only one comparator delay from the input, and the encoder N-bit output by only a few gate delays on top of that, so the process is very fast. However, the system uses large numbers of resistors and comparators for quite low resolutions, and if it is to be fast, each comparator must run at relatively high power levels. Hence, the problems of flash ADCs include limited resolution, high power dissipation (especially at sampling rates greater than 50MSPS), and relatively large (and therefore expensive) chip sizes. In addition, the resistance of the reference resistor chain must be kept low to supply adequate bias current to the fast comparators, so the voltage reference has to source quite large currents (>10 mA).

In practice, flash converters are available up to 10-bits (12 bit flash converters have been made, but are not commercially viable), but more commonly they have 8-bits of resolution. Their maximum sampling rate can be as high as 500 MSPS, and input full-power bandwidths in excess of 300 MHz are not uncommon.

But as we mentioned earlier, full-power bandwidths are not necessarily full-resolution bandwidths. Ideally, the comparators in a flash converter are well matched both for DC and AC characteristics. Because the strobe is

applied to all the comparators simultaneously, the flash converter is inherently a sampling converter. In practice, there are delay variations between the comparators and other AC mismatches which cause a degradation in ENOB at high input frequencies.

The input to a flash ADC is applied in parallel to a large number of comparators. Each has a voltage variable junction capacitance, and we must also consider the inductance of the long conductor tracks on the chip from the pad to the comparators. The combination of inductance and non-linear signal-dependent capacitance results in all flash ADCs having reduced ENOB at high input frequencies as shown in Figure 5.51, the exact amount of degradation will vary from type to type, but the basic problem is inherent in the architecture.

FLASH OR PARALLEL ADC

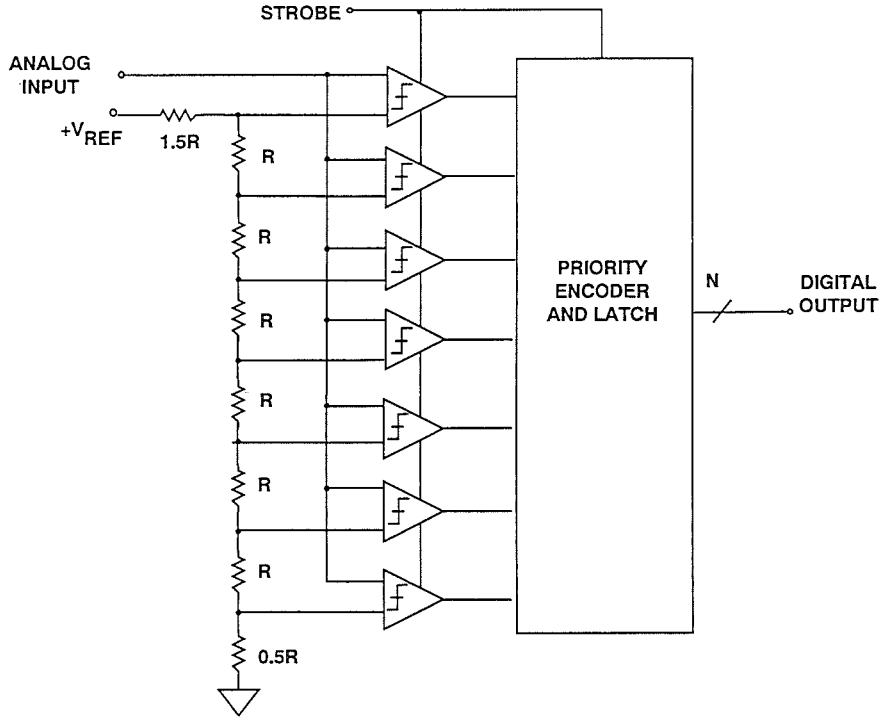


Figure 5.50

INPUT CIRCUIT MODEL FOR FLASH ADC AND ITS EFFECT ON DISTORTION

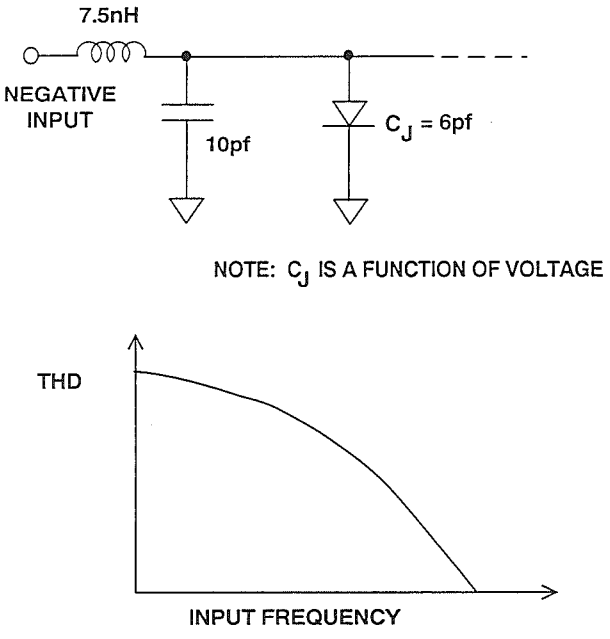


Figure 5.51

Subranging ADCs

Although it is not practical to make flash ADCs with high resolution, flash ADCs are used as subsystems in "subranging" ADCs (sometimes known as "half-flash ADCs"), which are capable of much higher resolutions (up to 16-bits).

Figure 5.52 shows a basic subranging architecture. The analog signal is applied to an N-bit flash converter. The result of the conversion is applied to an N-bit DAC with >2N-bit accuracy. The analog output from the DAC is subtracted from the original signal, and the remainder is amplified and applied to another N-bit flash ADC (or it could be switched to the same ADC that was used for the first conversion). The first conversion obtains the N MSBs. The D/A conversion and subtraction removes the MSB information from the analog signal, and the second A/D conversion provides the N LSBs.

SUBRANGING (HALF-FLASH) ADC

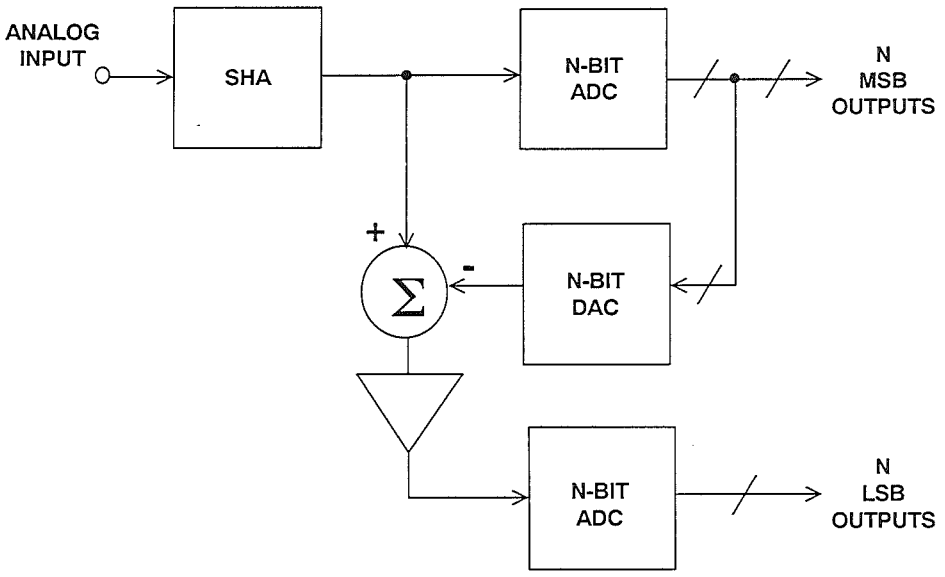


Figure 5.52

Obviously, such a procedure is not as fast as a simple flash conversion, but it is much faster than a successive approximation routine. Furthermore, the complexity and power for a given resolution is much less than for a flash converter. Subranging converters can have resolutions up to 16-bits and conversion rates of 50 MSPS (although not both together at the time of writing - mid 1994).

One of the limitations on the speed of a subranging converter is the settling time and the accuracy of the first A/D conversion and the D/A conversion. Although the first stage conversion is only N bits, it must be performed to greater than 2N bits of accuracy. Any mismatch between the two stages, or any error in the first conversion will show up as errors in the overall ADC transfer function. An alternate architecture which is known as *subranging with digital error correction* relaxes the accuracy and settling time requirements on the first stage and allows faster operation (see Figure 5.53).

SUBRANGING (HALF-FLASH) ADC WITH DIGITAL ERROR CORRECTION

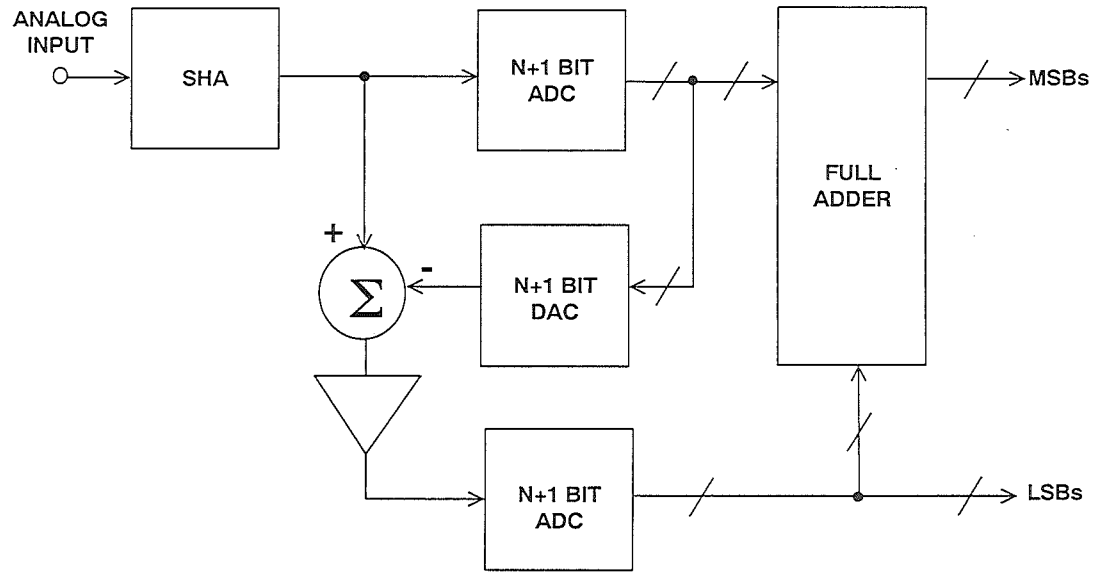


Figure 5.53

The basic architecture is unchanged, but the resolution of the two A/D conversions and the D/A conversion are increased by 1-bit. The first A/D conversion and the D/A conversion are performed more quickly, and there is some loss of accuracy due to inadequate settling time. However, the second A/D conversion is higher resolution and contains enough data to correct the error in the MSB conversion. The simplest way to do this is by use of a full adder. The increase in complexity is small, and the increase in speed and improvement in accuracy can be substantial.

So far, we have considered only two-stage subranging ADCs, as these are easiest to analyze. There is no reason to stop at two stages, however. Indeed, a successive approximation ADC might be considered the limiting case of a multi-stage subranging ADC, although we do not propose to analyze it in this way. However, three-pass and four-pass subranging ADCs are quite common, and can be made in many different ways, with or without digital error correction.

As examples of just two of the many arrangements possible, Figure 5.54 shows a block diagram of the AD1671, a 1.25 MSPS ADC using 4-pass conversion with digital error correction, two separate DACs and four separate flash ADCs. The AD779, which achieves 14-bits at 128 KSPS with a single 4-bit flash ADC and one DAC is shown in Figure 5.55. Study of Reference 4 will reveal examples of many other structures.

AD1671 12-BIT, 1.25MSPS MULTIPASS SUBRANGING ADC

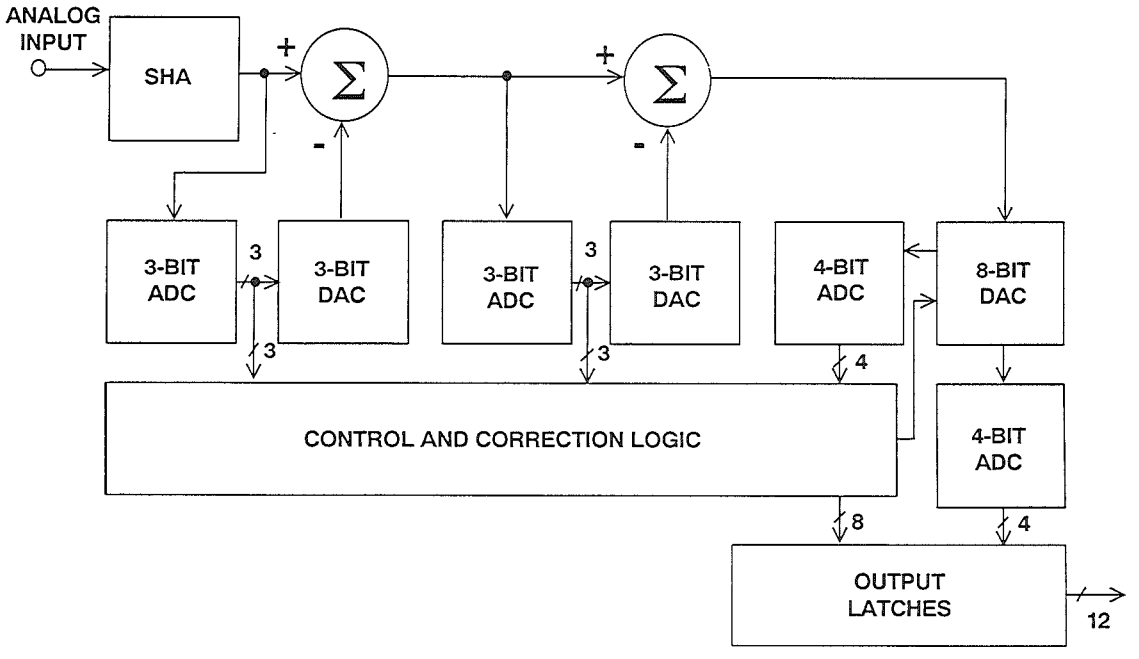


Figure 5.54

AD779 14-BIT, 128kSPS MULTIPASS (RECIRCULATING) ADC

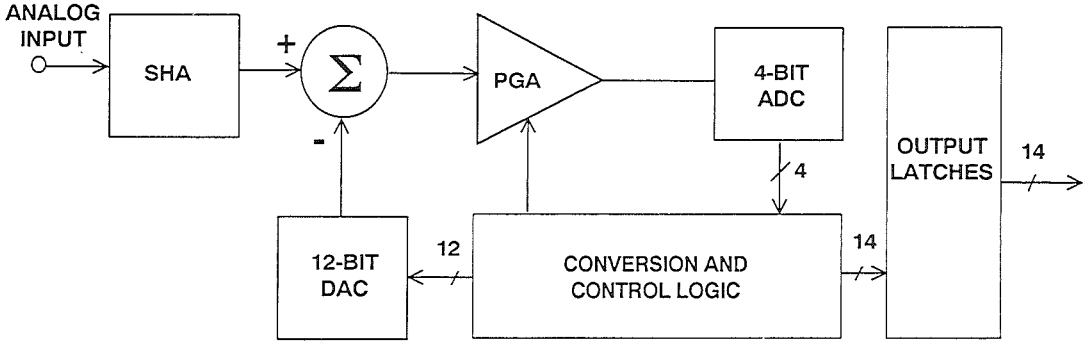


Figure 5.55

Integrating ADCs

Flash and subranging ADCs are fast, complex, power hungry, and are not inherently linear or free of missing codes. Integrating ADCs are simple, linear, very accurate, free of missing codes, and consume very little power, but they are slow (see Figure 5.56).

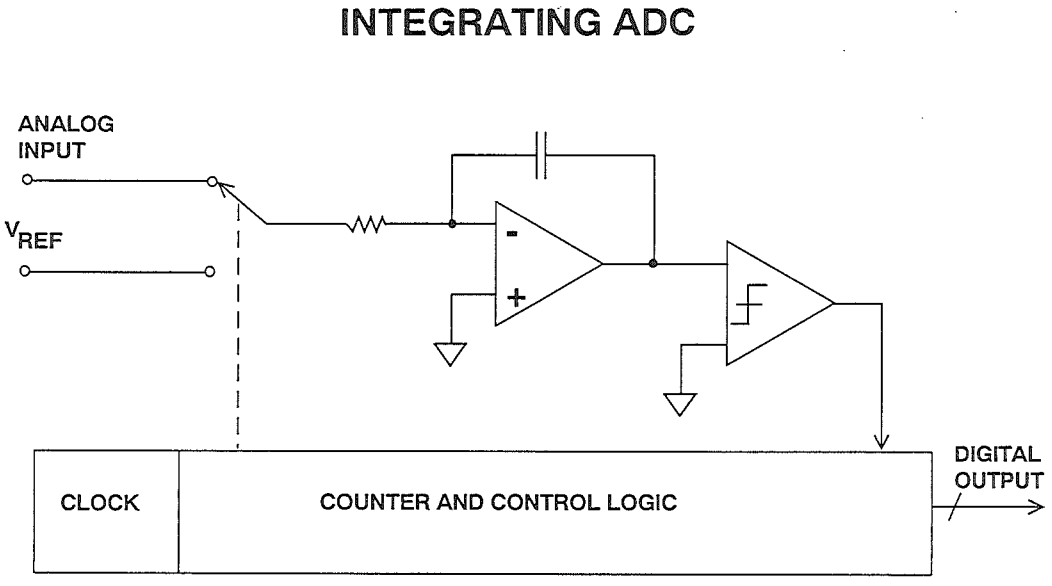


Figure 5.56

The integrator is charged from the unknown voltage for a fixed time defined by the converter clock. It is then fully discharged (connected to the reference voltage), and the discharge time measured by the same clock. The ratio of the input to the reference is the ratio of discharge to charge times. Since the integrator and the clock are the same during charge and discharge, they do not contribute to errors, although offset and bias currents do. However, these may be canceled out by using four charge/discharge cycles instead of two; the resulting conversion is very accurate.

Simple monolithic integrating ADCs using CMOS, which is not renowned for its analog amplifier or comparator performance, are quite capable of 16-bit accuracy. The technique is used in most DVMs (digital voltmeters) from portable 2½ digit ones operating for hundreds of hours from a pair of button cells, to 6½ digit transfer standards. Another advantage of the technique for DVM applications is that if the input changes during a measurement the result of the conversion is the mean value of the input over the conversion cycle. If a suitable conversion time is chosen, 50Hz or 60Hz line ripple will essentially be ignored by an integrating ADC (see Figure 5.57).

FREQUENCY RESPONSE OF INTEGRATING ADC

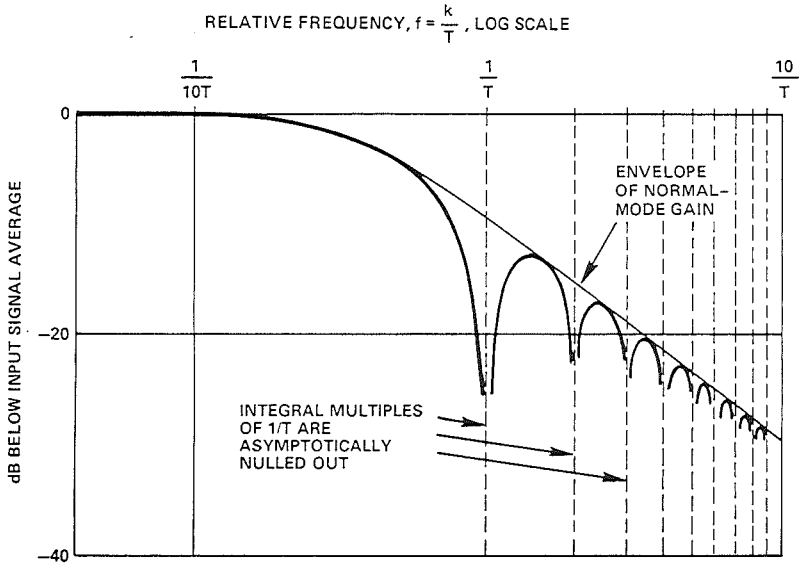


Figure 5.57

Voltage-to-Frequency Converter (VFC) / Counter ADCs

Another integrating ADC consists of a VFC (voltage to frequency converter) and a frequency counter.

A VFC is an oscillator whose frequency is linearly proportional to a control voltage. The VFC/counter ADC is monotonic and free of missing codes, integrates noise, and can consume very little power. It is also very useful for telemetry applications, since the VFC, which is small, cheap and low-powered can be mounted on the experimental subject (patient, wild animal, artillery shell, etc.) and communicate with the counter by a telemetry link.

VOLTAGE-TO-FREQUENCY CONVERTER (VFC) AND FREQUENCY COUNTER MAKE A LOW-COST, VERSATILE, HIGH-RESOLUTION ADC

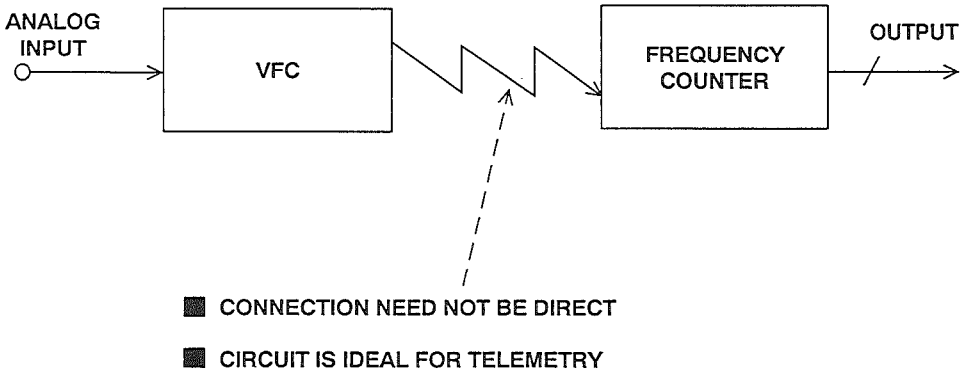


Figure 5.58

There are two common VFC architectures (there are many more VFO [variable frequency oscillator] architectures, including the ubiquitous 555, but the key feature of VFCs is linearity — few VFOs are very linear): the current steering multivibrator and the charge-balance VFC. The charge-balanced VFC may be made in asynchronous or synchronous (clocked) forms.

The current steering multivibrator is actually a current-frequency converter rather than a VFC, but, as shown in Figure 5.59, practical circuits invariably contain a voltage-current converter at the input. The principle of operation is evident: the current discharges the capacitor until a threshold is reached, and when the capacitor terminals are reversed, the half-cycle repeats itself. The waveform across the capacitor is a linear tri-wave, but the waveform on either terminal with respect to ground is the more complex waveform shown.

Practical VFCs of this type have linearities around 14-bits, and comparable stability, although they may be used in ADCs with higher resolutions without missing codes. The performance limits are set by comparator threshold noise, threshold temperature coefficient,

and the stability and dielectric absorption (DA) of the capacitor, which is generally a discrete component. The comparator/voltage reference structure shown in the diagram is more of a representation of the function performed than the actual circuit used, which is much more integrated with the switching, and correspondingly harder to analyze.

This type of VFC is simple, inexpensive, and low-powered, and most run from a wide range of supply voltages. They are ideally suited for low cost medium accuracy ADC and data telemetry applications.

The charge balance VFC is more complex, more demanding in its supply voltage and current requirements, and more accurate. It is capable of 16-18 bit linearity.

A CURRENT-STEERING VFC

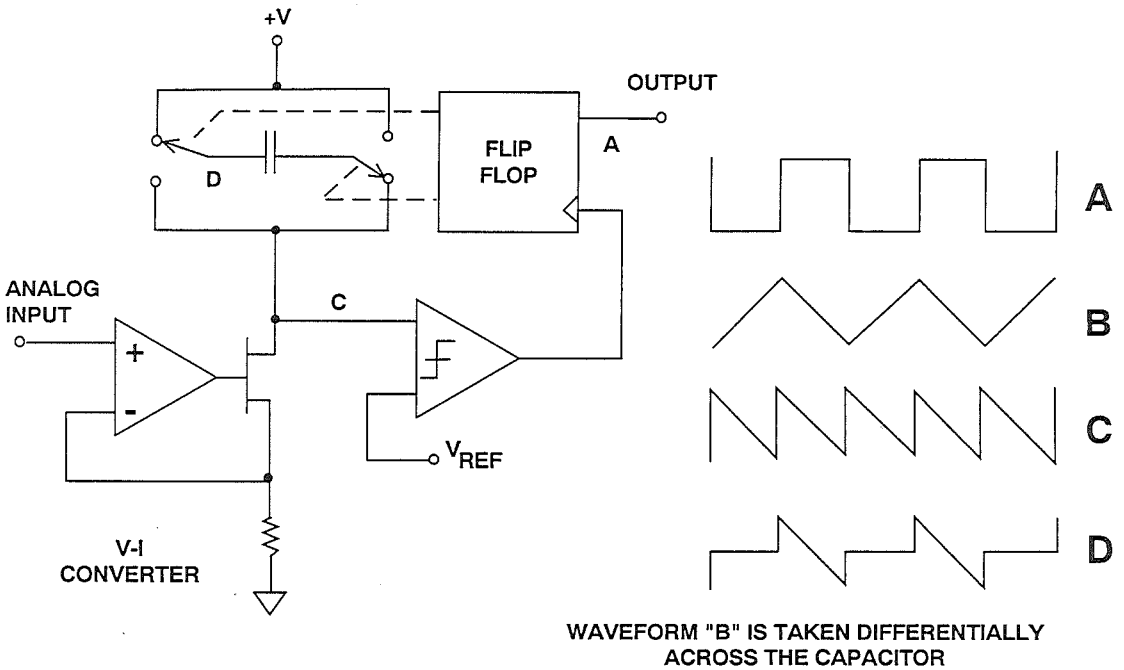


Figure 5.59

CHARGE-BALANCE VFC

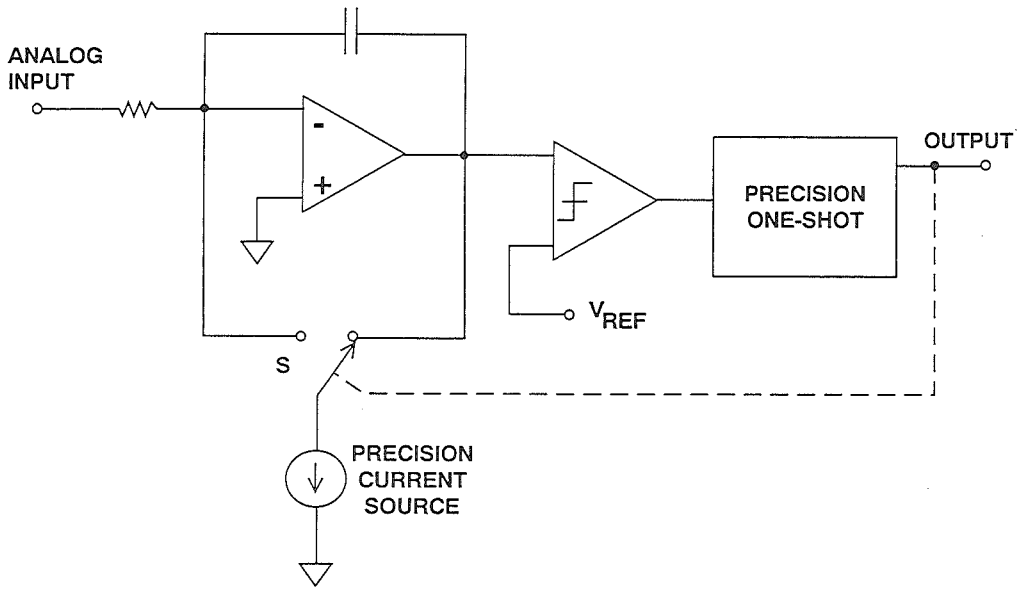


Figure 5.60

The integrator capacitor charges from the signal as shown in Figure 5.60. When it passes the comparator threshold, a fixed charge is removed from the capacitor, but the input current continues to flow during the discharge so no input charge is lost. The fixed charge is defined by the precision current source and the pulse width of the precision monostable. The output pulse rate is thus accurately proportional to the rate at which the integrator charges from the input.

At low frequencies, the limits on the performance of this VFC are set by the stability of the current source and the monostable timing (which depends on the monostable capacitor, among other things). The absolute value and temperature stability of the integration capacitor do not affect the accuracy, although its leakage and dielectric absorption (DA) do. At high frequencies, second-order effects, such as switching transients in the integrator and the precision of the monostable when it is retriggered very soon after the end of a pulse, take their toll on accuracy and linearity.

The changeover switch in the current source addresses the integrator transient problem. By using a changeover switch instead of the on/off switch more common on older VFC designs: (a) there are no on/off transients in the precision current source and (b) the output stage of the integrator sees a constant load — most of the time the current from the source flows directly in the output stage; during charge balance, it still

flows in the output stage, but through the integration capacitor.

The stability and transient behavior of the precision monostable present more problems, but the issue may be avoided by replacing the monostable with a clocked bistable multivibrator. This arrangement is known as a *synchronous* VFC or SVFC and is shown in Figure 5.61.

The difference from the previous circuit is quite small, but the charge balance pulse length is now defined by two successive edges of the external clock. If this clock has low jitter, the charge will be very accurately defined. The output pulse will also be synchronous with the clock. SVFCs of this type are capable of up to 18-bit linearity and excellent temperature stability.

SYNCHRONOUS VFC (SVFC)

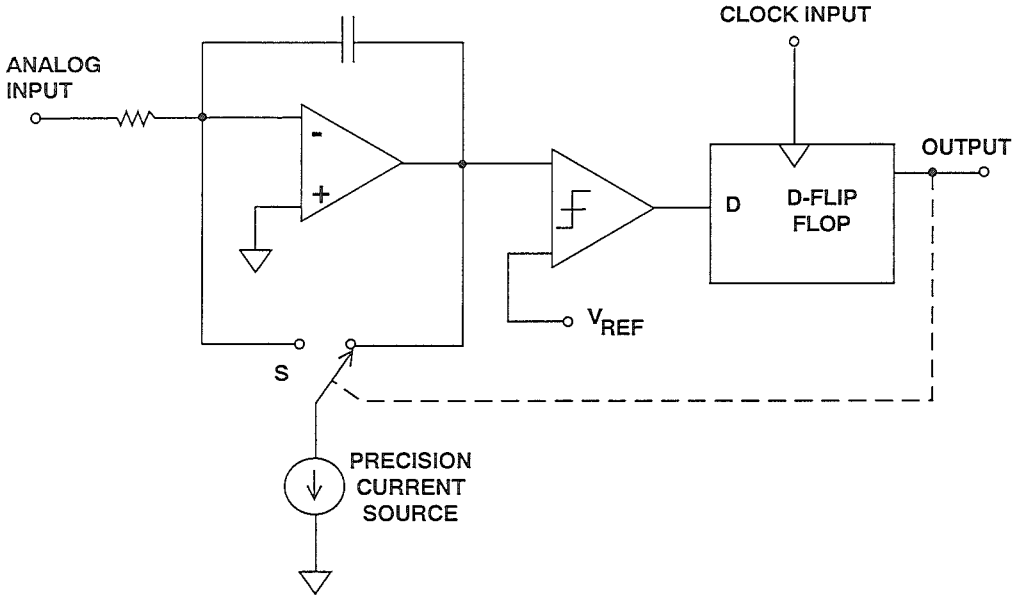


Figure 5.61

This synchronous behavior is convenient in many applications, since synchronous data transfer is often easier to handle than asynchronous. It does mean, however, that the output of an SVFC is not a pure tone (plus harmonics, of course) like a conventional VFC, but contains components harmonically related to the clock frequency. The display of an SVFC output on an oscilloscope is especially misleading and is a common cause of inquiries to the Applications Department: a change of input to a VFC produces a smooth change in

the output frequency, but a change to an SVFC produces a change in probability density of output pulses N and N+1 clock cycles after the previous output pulse, which is often perceived by inexperienced users as severe jitter and a sign of a faulty device (see Figure 5.62).

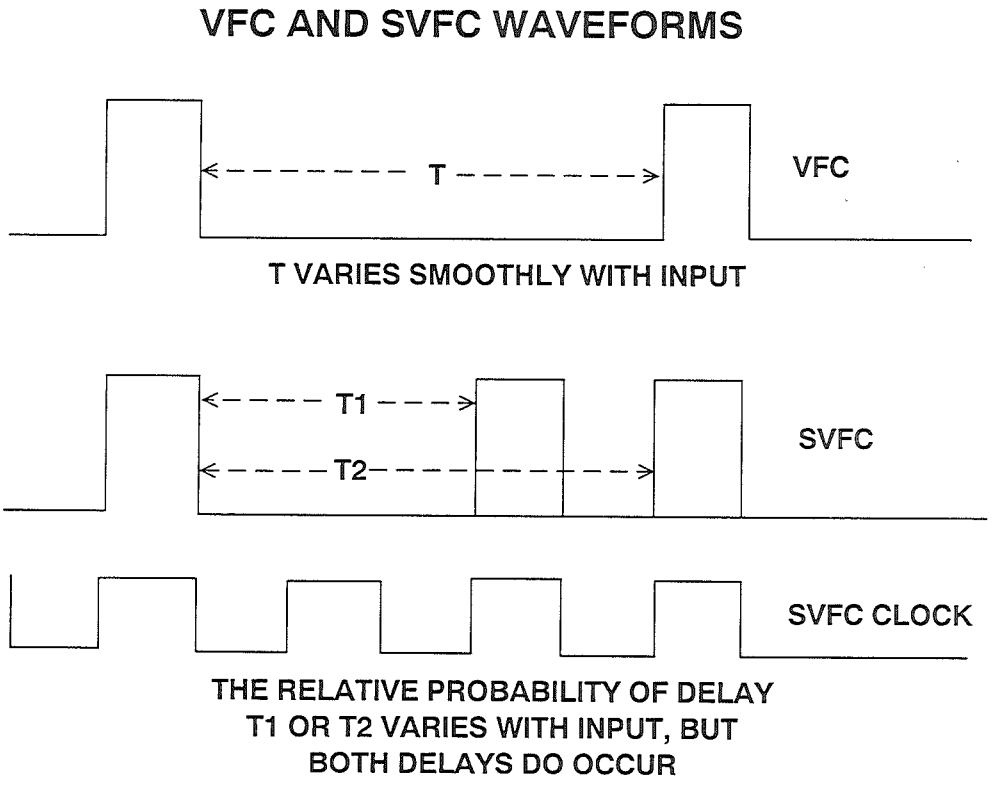


Figure 5.62

Another problem with SVFCs is non-linearity at output frequencies related to the clock frequency. If we study the transfer characteristic of an SVFC, we find non-linearities close to sub-harmonics of the clock frequency F_C as shown in Figure 5.63. They can be found at $F_C/3$, $F_C/4$, and $F_C/6$. This is due to stray capacitance on the chip (and in the circuit layout!) and coupling the clock signal into the SVFC comparator which causes the device to behave as an injection-locked phase-locked loop (PLL). This problem is intrinsic to SVFCs, but is not often serious: if the circuit card is well laid out, and clock amplitude and dV/dTs kept as low as practical, the effect is a discontinuity in the transfer characteristic of less than 8 LSBs (at 18-bit resolution) at $F_C/3$ and $F_C/4$, and less at other sub-harmonics. This is frequently tolerable, since the frequencies where it occurs are known. Of course, if the circuit layout or decoupling is poor, the effect may be much larger, but this is the fault of poor design and not the SVFC itself.

SVFC NON-LINEARITY

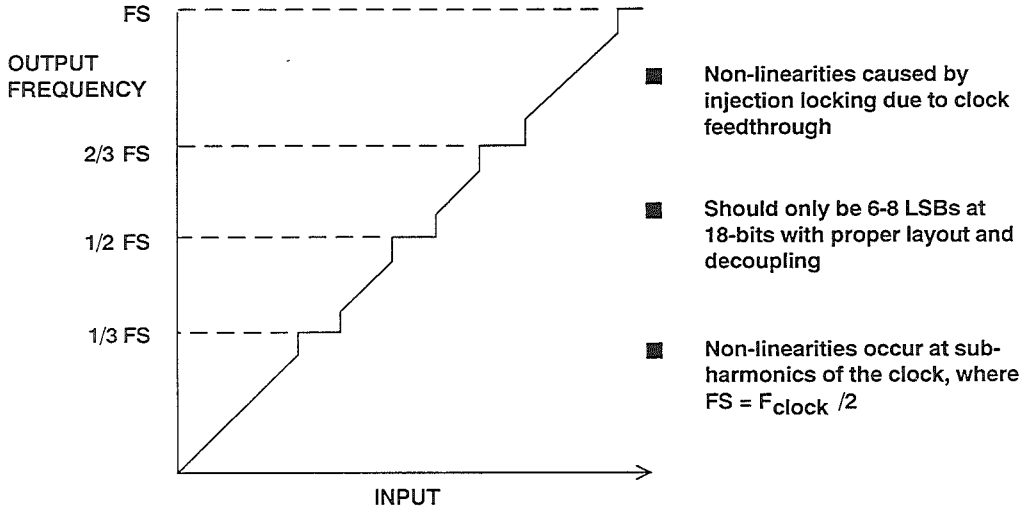


Figure 5.63

It is evident that the SVFC is quantized, while the basic VFC is not. It does NOT follow from this that the counter/VFC ADC has higher resolution (neglecting non-linearities) than the counter/SVFC ADC, because the clock in the counter also sets a limit to the resolution.

When a VFC has a large input, it runs quickly and (counting for a short time) gives good resolution, but it is hard to get good resolution in a reasonable sample time with a slow-running VFC. In such a case, it may be more practical to measure the period of the VFC output (this does not work for an SVFC), but of course the resolution of this system deteriorates as the input (and the frequency) increases. However, if the counter/timer arrangement is made smart, it is possible to measure the approximate VFC frequency and the exact period of not one, but N cycles (where the value of N is determined by the approximate frequency), and maintain high resolution over a wide range of inputs. The AD1170 is an example of this architecture.

VFCs have more applications than as a component in ADCs. Since their output is a pulse stream, it may easily be sent over a wide range of transmission media (PSN, radio, optical, IR, ultra-sonic, etc.). It need not be received by a counter, but by another VFC configured as a frequency-voltage converter (FVC). This gives an analog output, and a VFC-FVC combination is a very useful way of sending a precision analog signal across an isolation barrier. There are a number of issues to be considered in building FVCs from VFCs, and these are considered in Reference 5.

VFCs

- It is possible to use the PERIOD of a VFC, rather than its frequency, to measure the input signal.
- VFCs have other applications than as ADC elements. These include isolation and use as Frequency-to-Voltage Converters (FVCs).

Figure 5.64

Tracking ADCs

Having considered two types of fast ADCs and two types of slow ADCs, it is interesting to consider a type which is both fast and slow. The *tracking* ADC consists of a comparator, a DAC, and an up/down counter as shown in Figure 5.65.

TRACKING ADC

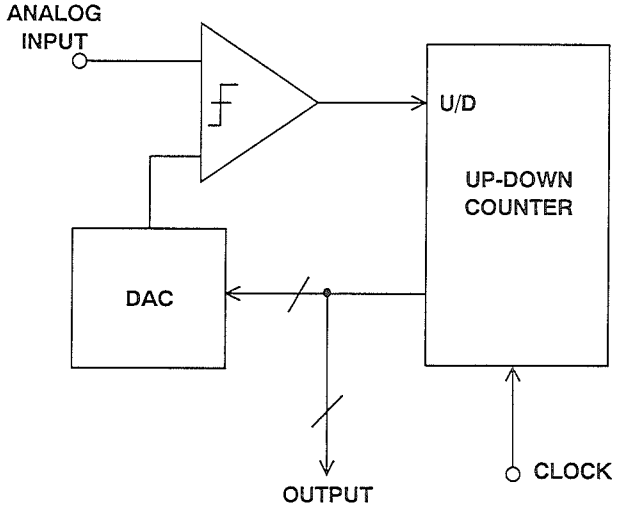


Figure 5.65

The analog input is applied to one input of the comparator, and the DAC output to the other. The DAC input is driven by the counter. If the analog input exceeds the DAC output, the counter counts up until they are equal. If the DAC output exceeds the analog input, the counter counts down until they are equal. It is evident that if the analog input changes slowly, the counter will follow, and the digital output will remain close to its correct value, whereas if the analog input suddenly undergoes a large step change, it will be many hundreds or thousands of clock cycles before the output is again valid - hence the statement that such an ADC is both fast and slow: it responds quickly to a slowly changing signal, but slowly to a quickly changing one.

The simple analysis above ignores the behavior of the ADC when the analog input and DAC output are nearly equal. This will depend on the exact nature of the comparator and counter. If the comparator is a simple one, the DAC output will cycle by 1 LSB from just above the analog input to just below it, and the digital output will, of course, do the same - there will be 1 LSB of flicker. Note that the output in such a case steps every clock cycle, irrespective of the exact value of analog input, and hence always has unity Mark/Space ratio. In other words, there is no possibility of taking a mean value of the digital output and increasing resolution by oversampling.

A more satisfactory, but more complex arrangement would be to use a window comparator with a window 1-2 LSB wide. When the DAC output is high or low the system behaves as in the previous description, but if the DAC output is within the window, the counter stops. This arrangement eliminates the flicker, provided that the DAC DNL never allows the DAC output to step across the window for 1 LSB change in code.

Tracking ADCs are not very common. Their slow step response makes them unsuitable for many applications, and they have few compensating advantages. But they do have one asset: their output is *continuously* available. Most ADCs perform conversions: i.e., on receipt of a "start convert" command (which may be internally generated), they perform a conversion and, after a delay, a result becomes available. Providing that the analog input changes slowly, the output of a tracking ADC is always available. This is valuable in synchro-to-digital and resolver to digital converters (SDCs and RDCs), and this is the application where tracking ADCs are most often used. A very small change in their architecture, however, produces one of the most widespread ADC designs, the successive approximation ADC.

Successive Approximation ADCs

In a successive approximation ADC (see Figure 5.66), the counter of the tracking ADC is replaced by a successive approximation register (SAR). The successive approximation ADC performs conversions on command. On the START CONVERT command, all the bits of the SAR are reset to "0" except the MSB which is set to "1". If the DAC output is greater than the analog input, this bit is reset, otherwise it is left set. The next most significant bit is then set to "1". If the DAC output is greater than the analog input, this bit is reset, otherwise it is left set. The process is repeated with each bit in turn. When all the bits have been set, tested, and reset or not as appropriate, the contents of the SAR correspond to the value of the analog input, and the conversion is complete.

An N-bit conversion takes N steps. It would seem on superficial examination that a 16-bit converter would have one-half the conversion rate of an 8-bit one, but this is not the case. In an 8-bit converter, the DAC must settle to 8-bit accuracy before the bit decision is made, whereas in a 16-bit converter, it must settle to 16-bit accuracy, which takes a lot longer. In practice, 8-bit successive approximation ADCs can convert in a few hundred nanoseconds, while 16-bit ones will generally take several microseconds.

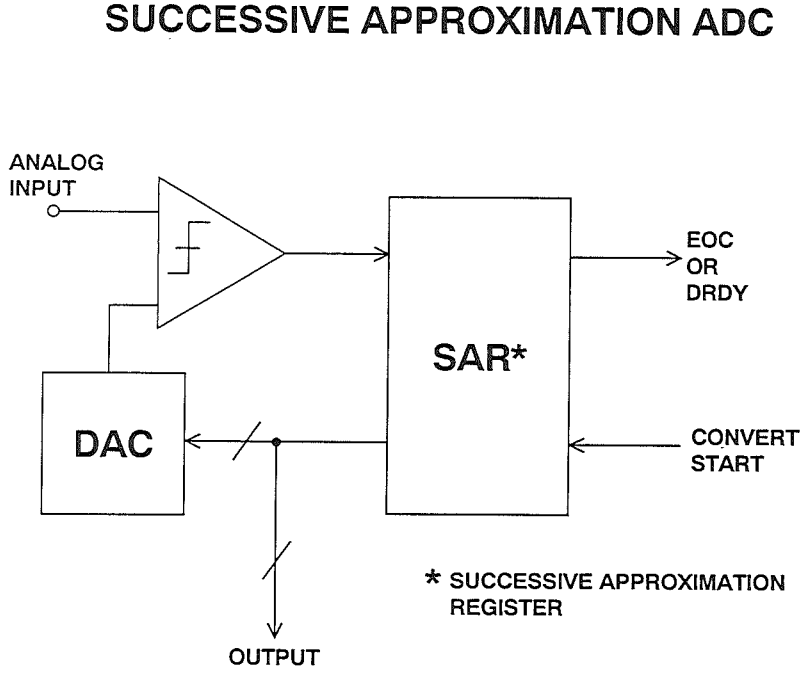


Figure 5.66

The successive approximation ADC has a very simple structure, is low power, and has reasonably fast conversion times. It is probably the most widely used ADC architecture, and will continue to be used for medium speed and medium resolution applications. As levels of analog integration increase, however, the subranging ADC will be increasingly used for high-speed, medium resolution applications, and the rise of cheap digital signal processing (DSP) makes the sigma-delta architecture increasingly attractive for low bandwidths and very high resolutions. However, its theory is complex, and its mode of operation often misunderstood.

Sigma-Delta (Σ–Δ) ADCs

Sigma-Delta Analog-Digital Converters (Σ–Δ ADCs) have been known for nearly thirty years, but only recently has the technology (high-density digital VLSI) existed to manufacture them as inexpensive monolithic integrated circuits. They are now used in many applications where a low-cost, low-bandwidth, high-resolution ADC is required.

There have been innumerable descriptions of the architecture and theory of Σ–Δ ADCs, but most commence with a maze of integrals and deteriorate from there. In the Applications Department at Analog Devices, we frequently encounter engineers who do not understand the theory of operation of Σ–Δ ADCs and are convinced, from study of a typical published article, that it is too complex to comprehend easily.

There is nothing particularly difficult to understand about Σ–Δ ADCs, as long as you avoid the detailed mathematics, and this section has been written in an attempt to clarify the subject. A Σ–Δ ADC contains very simple analog electronics (a comparator, a switch, and one or more integrators and analog summing circuits), and quite complex digital computational circuitry. This circuitry consists of a digital signal processor (DSP) which acts as a filter (generally, but not invariably, a low pass filter). It is not necessary to know how the filter works to appreciate what it does. To understand how a Σ–Δ ADC works one should be familiar with the concepts of *over-sampling, noise shaping, digital filtering.* and *decimation*.

SIGMA-DELTA (Σ-Δ) ADCs

- Sigma-Delta ADCs are low-cost and have high resolution, excellent DNL, although limited input bandwidth
- A Σ-Δ ADC is Simple
- The Mathematics, however, is Complex
- This section concentrates on What Actually Happens!

Figure 5.67

SIGMA-DELTA ADC KEY CONCEPTS

- Oversampling
- Noise Shaping
- Digital Filtering
- Decimation

Figure 5.68

As we have said earlier in this section, an ADC is a circuit whose digital output is proportional to the ratio of its analog input to its analog reference. Often, but by no means always, the scaling factor between the analog reference and the analog signal is unity, so the digital signal represents the normalized ratio of the two.

Figure 5.69 (which we have seen before) shows the transfer characteristic of such a 3-bit unipolar ADC. The input to an ADC is analog and is not quantized, but its output is quantized. The transfer characteristic therefore consists of eight horizontal steps (when considering the offset, gain and linearity of an ADC we consider the line joining the midpoints of these steps).

Digital full scale (all "1"s) corresponds to 1 LSB below the analog full scale (the reference or some multiple thereof). This is because, as mentioned above, the digital code represents the *normalized* ratio of the analog signal to the reference, and if this were unity, the digital code would be all "0"s and "1" in the bit *above* the MSB.

The (ideal) ADC transitions take place at ½ LSB above zero and thereafter every LSB, until 1½ LSB below analog full scale. Since the analog input to an ADC can take any value, but the digital output is quantized, there may be a difference of up to ½ LSB between the actual analog input and the exact value of the digital output. This is known as the *quantization error* or *quantization uncertainty*. In AC (sampling) applications, this quantization error gives rise to *quantization noise*.

TRANSFER CHARACTERISTIC OF A 3-BIT UNIPOLAR ADC

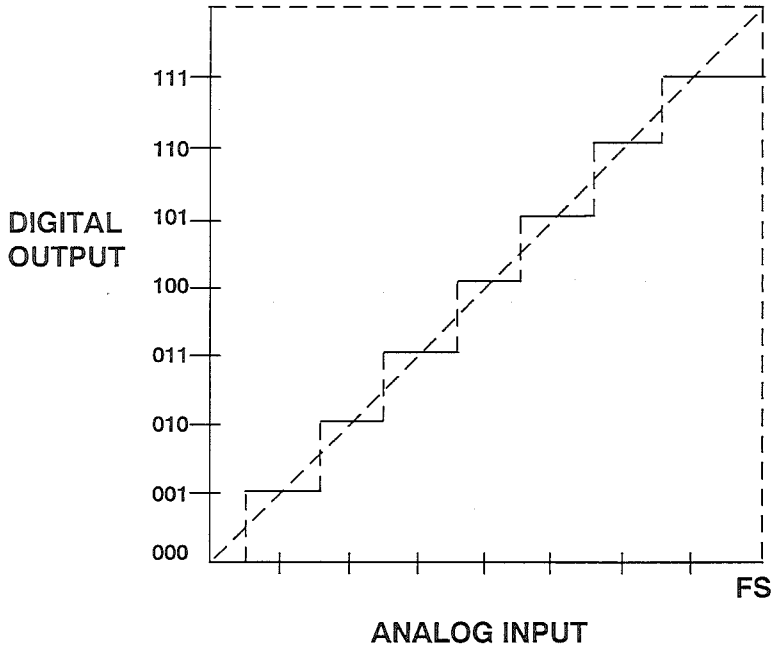


Figure 5.69

If we apply a fixed input to an ideal ADC, we will always obtain the same output, and the resolution will be limited by the quantization error.

Suppose, however, that we add some AC (dither) to the fixed signal, take a large number of samples, and prepare a histogram of the results. We will obtain something like the result in Figure 5.70. If we calculate the mean value of a large number of samples, we will find that we can measure the fixed signal with greater resolution than that of the ADC we are using. This procedure is known as *over-sampling*.

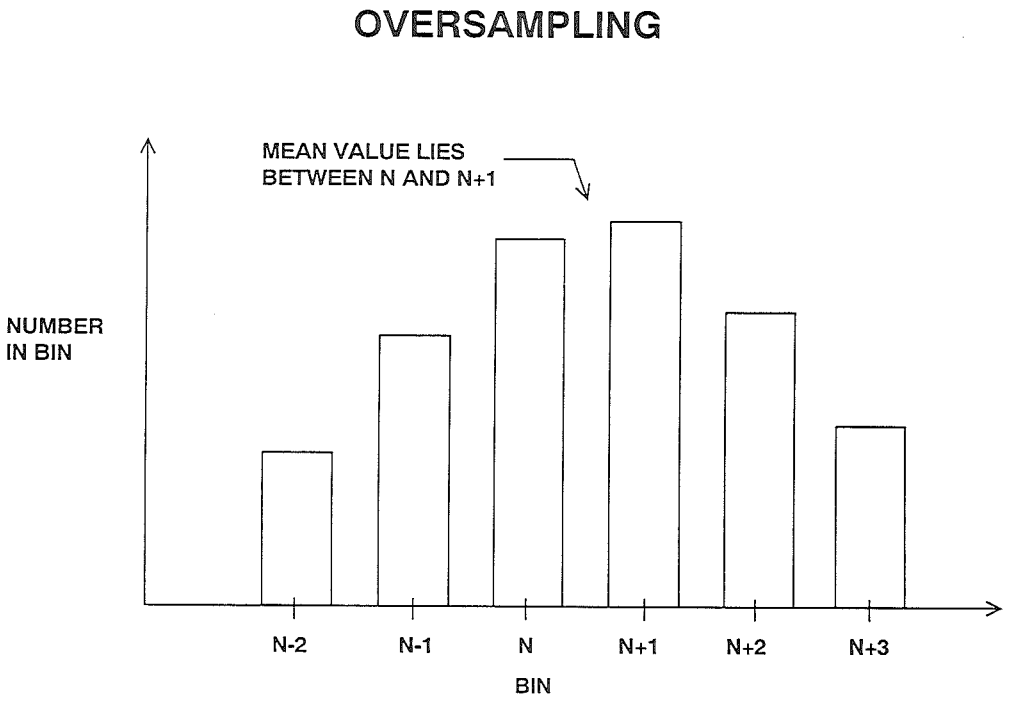


Figure 5.70

The AC (dither) that we add may be a sine-wave, a tri-wave, or gaussian noise (but *not* a square wave) and, with some types of sampling ADCs (including Σ–Δ ADCs), an external dither signal is unnecessary, since the ADC generates its own. Analysis of the effects of differing dither waveforms and amplitudes is complex and, for the purposes of this section, unnecessary. What we do need to know is that with the simple over-sampling described here, the number of samples must be doubled for each bit of increase in resolution.

If, instead of a fixed DC signal, the signal that we are over-sampling is an AC signal, then it is not necessary to add a dither signal to it in order to over-sample, since the signal is moving anyway. (If the AC signal is a single tone harmonically related to the sampling frequency, dither may be necessary, but this is a special case.)

Let us consider the technique of over-sampling with an analysis in the frequency domain. Where a DC conversion has a *quantization error* of up to ½ LSB,

a sampled data system has *quantization noise*. As we have already seen, a perfect classical N-bit sampling ADC has an rms quantization noise of $q/\sqrt{12}$ uniformly distributed within the Nyquist band of DC - $f_S/2$ (where q is the value of an LSB and f_S is the sampling rate). Therefore, its SNR with a full-scale sinewave input will be (6.02N + 1.76) dB. If the ADC is less than perfect, and its noise is greater than its theoretical minimum quantization noise, then its *effective* resolution will be less than N-bits. Its actual resolution (often known as its Effective Number of Bits or ENOB) will be defined by

$$\text{ENOB} = \frac{\text{SNR} - 1.76\text{dB}}{6.02\text{dB}}.$$

SAMPLING ADC QUANTIZATION NOISE

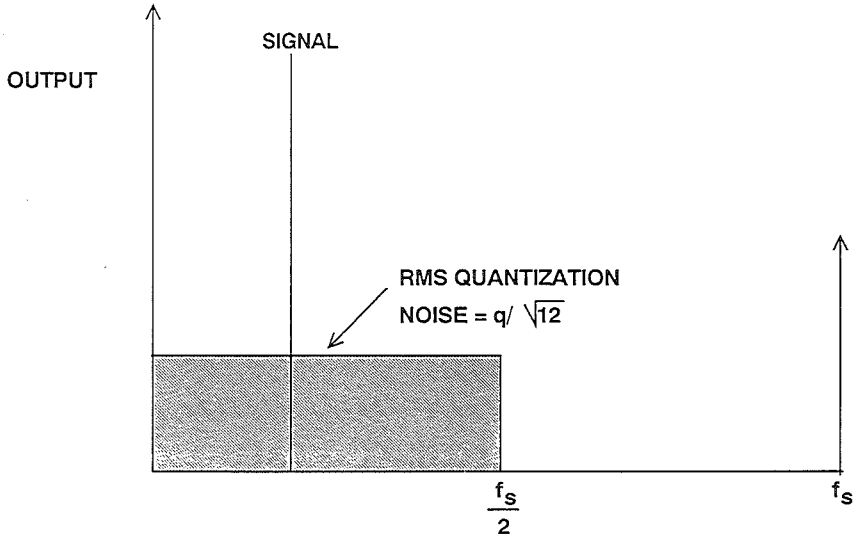


Figure 5.71

If we choose a much higher sampling rate, the quantization noise is distributed over a wider bandwidth as shown in Figure 5.72. If we then apply a digital low pass filter (LPF) to the output, we remove much of the quantization noise, but do not affect the wanted signal - so the ENOB is improved. We have accomplished a high resolution A/D conversion with a low resolution ADC.

OVERSAMPLING FOLLOWED BY DIGITAL FILTERING AND DECIMATION IMPROVES SNR AND ENOB

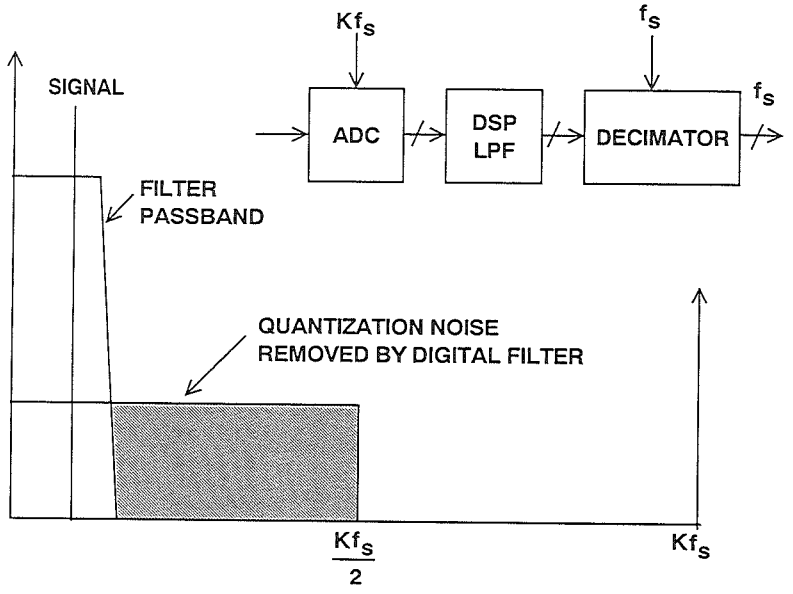


Figure 5.72

Since the bandwidth is reduced by the digital output filter, the output data rate may be lower than the original sampling rate and still satisfy the Nyquist criterion. This may be achieved by passing every Mth result to the output and discarding the remainder. The process is known as "decimation" by a factor of M. Despite the origins of the term (*decem* is Latin for ten), M can have any integer value, provided that the output data rate is more than twice the signal bandwidth. Decimation does not cause any loss of information (see Figure 5.73).

If we simply use over-sampling to improve resolution, we must over-sample by a factor of 2^N to obtain an N-bit increase in resolution. The Σ–Δ converter does not need such a high over-sampling ratio because it not only limits the signal passband, but also shapes the quantization noise so that it falls outside this passband.

If we take a 1-bit ADC (generally known as a comparator), drive it with the output of an integrator, and feed the integrator with an input signal summed with the output of a 1-bit DAC fed from

DECIMATION

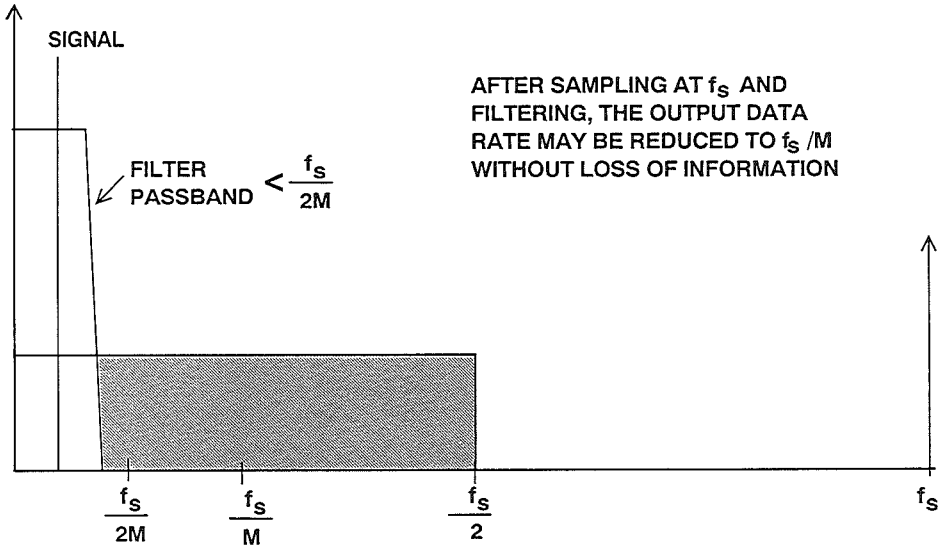


Figure 5.73

FIRST-ORDER SIGMA-DELTA ADC

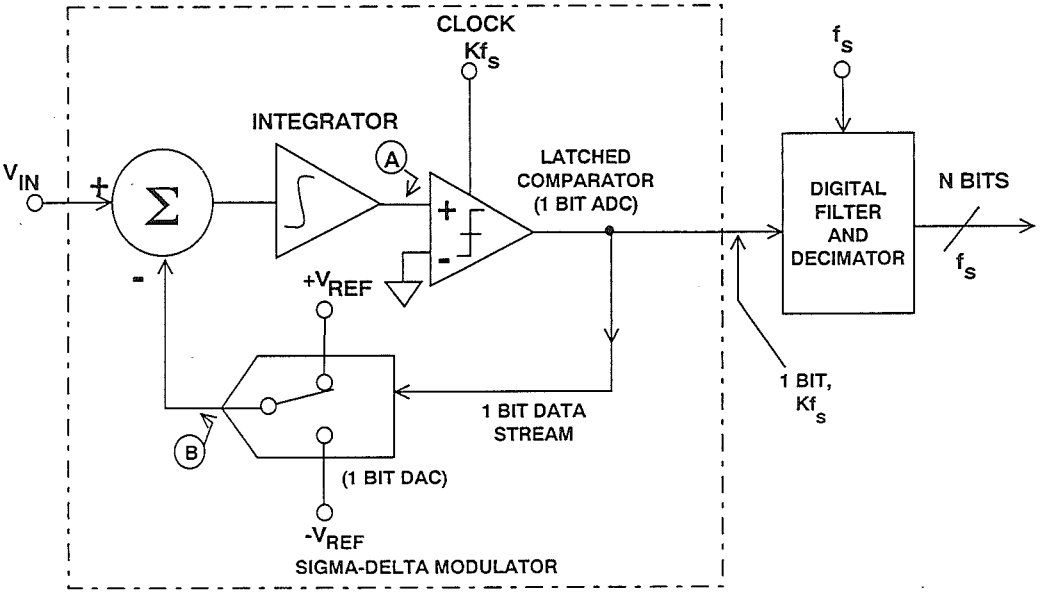


Figure 5.74

the ADC output, we have a first-order Σ–Δ modulator as shown in Figure 5.74. Add a digital low pass filter (LPF) and decimator at the digital output, and we have a Σ–Δ ADC: the Σ–Δ modulator shapes the quantization noise so that it lies above the passband of the output filter, and the ENOB is therefore much larger than would otherwise be expected from the over-sampling ratio.

By using more than one integration and summing stage in the Σ–Δ modulator, we can achieve higher orders of quantization noise shaping and even better ENOB for a given over-sampling ratio as is shown in Figure 5.75 for a second-order Σ–Δ modulator. The block diagram for the second-order Σ–Δ modulator is shown in Figure 5.76. Third, and higher, order Σ–Δ ADCs were once thought to be potentially unstable at some values of input — recent analyses using *finite* rather than infinite gains in the comparator have shown that this is not necessarily so, but even if instability does start to occur, it is not important, since the DSP in the digital filter and decimator can very easily recognize incipient instability and react to prevent it.

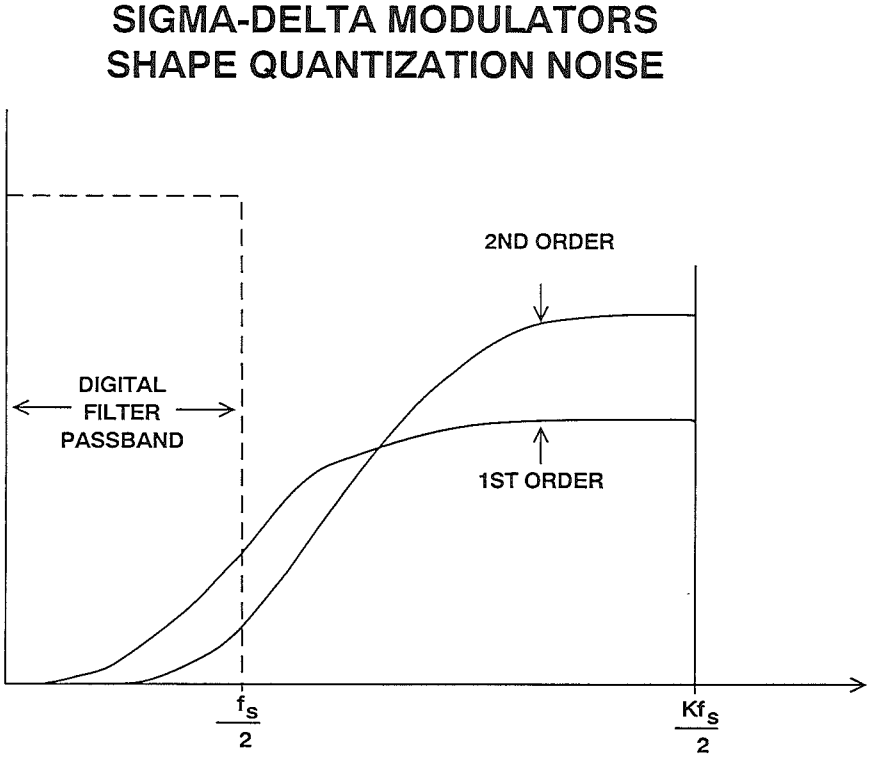


Figure 5.75

SECOND-ORDER SIGMA-DELTA ADC

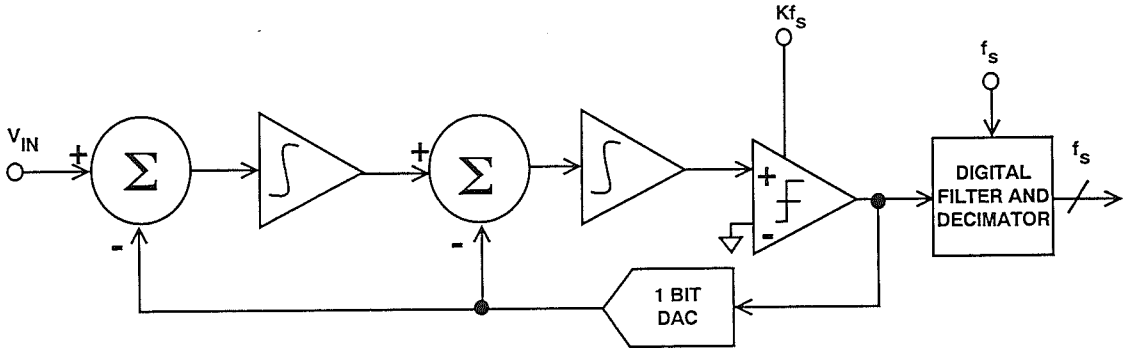


Figure 5.76

Figure 5.77 shows the relationship between the order of the Σ–Δ modulator and the amount of over-sampling necessary to achieve a particular SNR.

SNR VERSUS OVERSAMPLING RATIO FOR FIRST, SECOND, AND THIRD-ORDER LOOPS

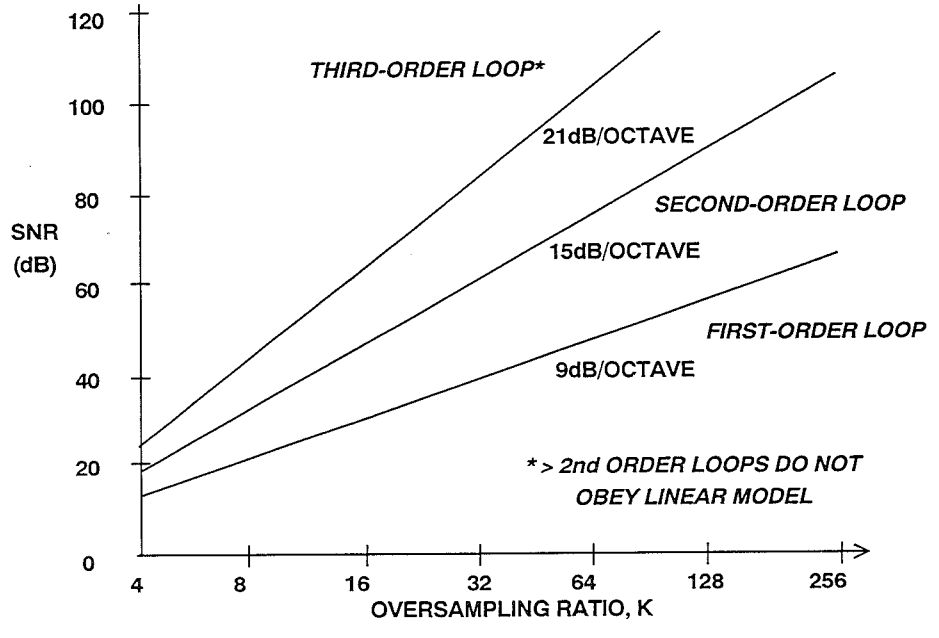


Figure 5.77

The Σ–Δ ADCs that we have described so far contain integrators, which are low pass filters, whose passband extends from DC. Thus, their quantization noise is pushed up in frequency. At present, all commercially available Σ–Δ ADCs are of this type (although some which are intended for use in audio or telecommunications applications contain bandpass rather than lowpass digital filters to eliminate any system DC offsets). Σ–Δ ADCs are available with resolutions up to 24-bits for DC measurement applications (AD7710), and with resolutions of 18-bits for high quality digital audio applications (AD1879).

But there is no particular reason why the filters of the Σ–Δ modulator should be LPFs, except that traditionally ADCs have been thought of as being baseband devices, and that integrators are somewhat easier to construct than bandpass filters. If we replace the integrators in a Σ–Δ ADC with bandpass filters (BPFs), the quantization noise is moved up and down in frequency to leave a virtually noise-free region in the pass-band (see Reference 6). If the digital filter is then programmed to have its pass-band in this region, we have a Σ–Δ ADC with a bandpass, rather than a low pass characteristic (see Figure 5.78). Although studies of this architecture are in their infancy, such ADCs would seem to be ideally suited for use in digital radio receivers, medical ultrasound, and a number of other applications.

REPLACING INTEGRATORS WITH RESONATORS GIVES A BANDPASS SIGMA-DELTA ADC

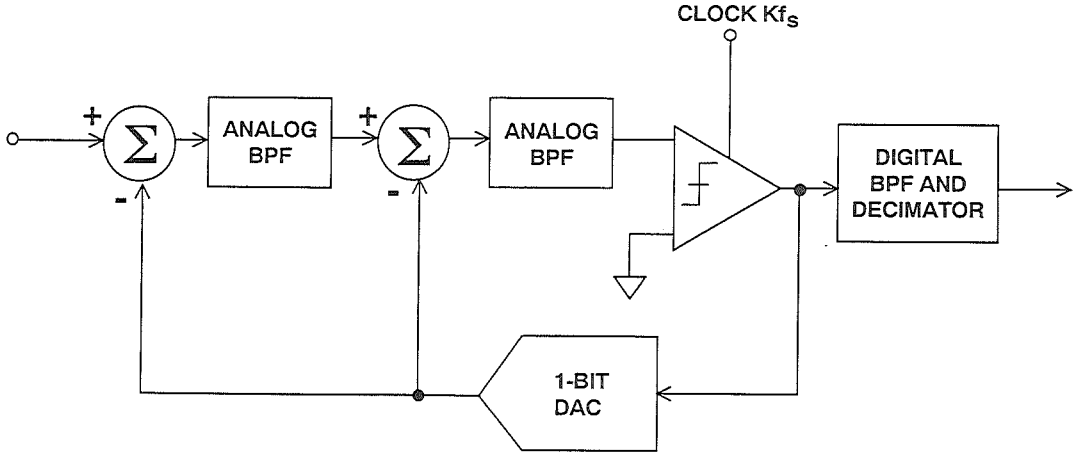


Figure 5.78

A Σ–Δ ADC works by over-sampling, where simple analog filters in the Σ–Δ modulator shape the quantization noise so that the SNR *in the bandwidth of interest* is much lower than would otherwise be the case, and by using high performance digital filters and decimation to eliminate noise outside the required passband. Because the analog circuitry is so simple and undemanding, it may be built with the same digital VLSI process that is used to fabricate the DSP circuitry of the digital filter. Because the basic ADC is 1-bit (a comparator), the technique is inherently linear.

Although the detailed analysis of Σ–Δ ADCs involves quite complex mathematics, their basic design can be understood without the necessity of any mathematics at all. For further discussion on Σ–Δ ADCs, refer to References 7 and 8.

SIGMA-DELTA SUMMARY

- Inherently Excellent Linearity
- Ideal for Mixed-Signal IC Processes, no Trimming
- No SHA Required
- Upper Sampling Rate Currently Limits Applications to Measurement, Voiceband, and Audio, but Bandpass Sigma-Delta Techniques Will Change This
- Analog Multiplexing Applications are Limited by Internal Filter Settling Time. Consider One Sigma-Delta ADC per Channel.

Figure 5.79

Sample and Hold Amplifiers (SHAs)

If successive approximation and subranging ADCs are to work correctly, the analog input must remain constant to within ±½ LSB during conversion. If we calculate the constraints that this imposes on the maximum input frequency, it is alarming: the maximum full-scale sinusoidal input frequency to a 10μs, 16-bit ADC is 0.24 Hz - less than one cycle every four seconds. The Nyquist frequency of such a converter, however, is 50kHz.

THE ADDITION OF AN EXTERNAL WIDEBAND LOW DISTORTION SHA EXTENDS THE LOW FREQUENCY PERFORMANCE OF THE ADC TO HIGHER FREQUENCIES

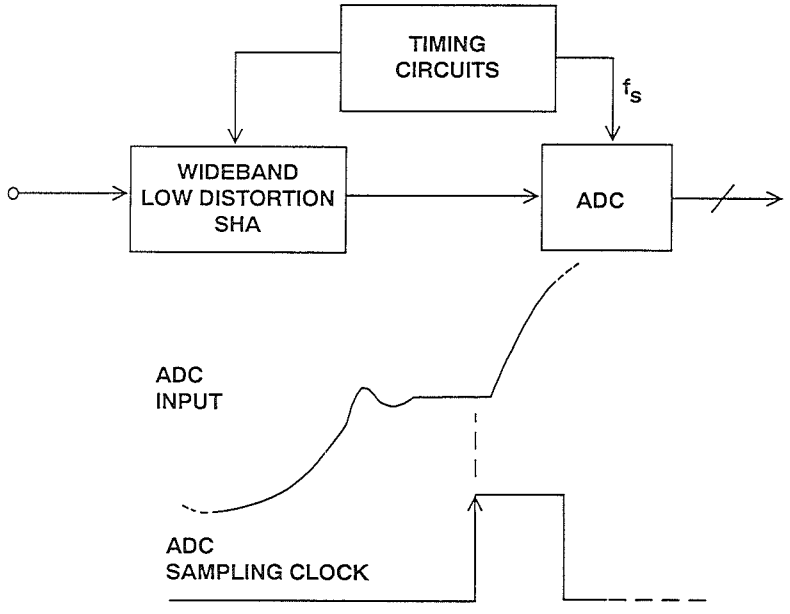


Figure 5.80

To overcome the problem (and also to improve the ENOB of some flash ADCs), all that is necessary is to incorporate a sample-and-hold (SHA) in the system as we have already mentioned. A SHA is a circuit which latches an analog signal in the state that it was at the moment a "sample" command was received. Sample-and-hold circuits are often referred to as "track-and-hold" circuits (THAs, or T/Hs). For most purposes the two terms are interchangeable, though there is a difference: the output of a SHA does not need to track the analog input at any time, whereas a THA must track as well as hold. For ADC input applications, the track function is not necessary, but

there are other places where it is. SHAs are discussed in detail in Section 8 of this book and Reference 9, and it is not necessary to analyze them here.

Most modern ADCs incorporate the SHA function on chip. This is usually far better than using a separate SHA, since the parameters of the SHA will be tailored to the needs of the ADC, and the whole system will be specified, rather than having separate SHA and ADC specifications which must be married together.

In general, *sampling* ADCs with integrated SHAs should be used wherever the system design requires the use of a SHA. One exception is with flash ADCs, since most flash ADCs are built with processes that are incapable of making a good SHA. Another is where the ENOB of an ADC working at a low sampling rate and a high input frequency ("undersampling") can be improved by a better external SHA.

REFERENCES

1. **Mixed Signal Design Seminar**, Analog Devices, Inc., 1991.

2. **System Applications Guide**, Analog Devices, Inc., 1994.

3. **Amplifier Applications Guide**, Analog Devices, Inc., 1992, pp. X-20 - X-35.

4. **Data Converter Reference Manual, Vol. II**, Analog Devices, Inc., 1992.

5. James M. Bryant, *Frequency-Voltage Converters*, Analog Devices, Inc., Application Note.

6. S.A.Jantzi, M.Snelgrove & P.F.Ferguson Jr., *A 4th-Order Bandpass Sigma-Delta Modulator*, **IEEE Journal of Solid State Circuits,** Vol. 38, No. 3, March 1993, pp.282-291.

7. **System Applications Guide**, Analog Devices, Inc., 1994, Section 14.

8. **Mixed Signal Design Seminar**, Analog Devices, Inc., 1991, Section 6.

9. Daniel H. Sheingold (Ed.), **Analog-Digital Conversion Handbook**, Third Edition, Prentice-Hall, 1986.