

Understanding Data Converters

SLAA013
July 1995



Printed on Recycled Paper

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

Contents

<i>Section</i>	<i>Title</i>	<i>Page</i>
1	Introduction	1
2	The Ideal Transfer Function	1
2.1	Analog-to-Digital Converter (ADC)	1
2.2	Digital-to-Analog Converter (DAC)	3
3	Sources of Static Error	4
3.1	Offset Error	4
3.2	Gain Error	5
3.3	Differential Nonlinearity (DNL) Error	6
3.4	Integral Nonlinearity (INL) Error	7
3.5	Absolute Accuracy (Total) Error	8
4	Aperture Error	9
5	Quantization Effects	10
6	Ideal Sampling	12
7	Real Sampling	13
8	Aliasing Effects and Considerations	14
8.1	Choice of Filter	14
8.2	Types of Filter	14
8.2.1	Butterworth Filter	15
8.2.2	Chebyshev Filter	15
8.2.3	Inverse Chebyshev Filter	15
8.2.4	Cauer Filter	15
8.2.5	Bessel-Thomson Filter	15
8.3	TLC04 Anti-Aliasing Butterworth Filter	16

List of Illustrations

<i>Figure</i>	<i>Title</i>	<i>Page</i>
1.	The Ideal Transfer Function (ADC)	2
2.	The Ideal Transfer Function (DAC)	3
3.	Offset Error	4
4.	Gain Error	5
5.	Differential Nonlinearity (DNL)	6
6.	Integral Nonlinearity (INL) Error	7
7.	Absolute Accuracy (Total) Error	8
8.	Aperture Error	9
9.	Quantization Effects	10
10.	Ideal Sampling	12
11.	Real Sampling	13
12.	Aliasing Effects and Considerations	14
13.	TLC04 Anti-aliasing Butterworth Filter	16

1 INTRODUCTION

This application report discusses the way the specifications for a data converter are defined on a manufacturer's data sheet and considers some of the aspects of designing with data conversion products. It covers the sources of error that change the characteristics of the device from an ideal function to reality.

2 THE IDEAL TRANSFER FUNCTION

The theoretical ideal transfer function for an ADC is a straight line, however, the practical ideal transfer function is a uniform staircase characteristic shown in Figure 1. The DAC theoretical ideal transfer function would also be a straight line with an infinite number of steps but practically it is a series of points that fall on the ideal straight line as shown in Figure 2.

2.1 Analog-to-Digital Converter (ADC)

An ideal ADC uniquely represents all analog inputs within a certain range by a limited number of digital output codes. The diagram in Figure 1 shows that each digital code represents a fraction of the total analog input range. Since the analog scale is continuous, while the digital codes are discrete, there is a quantization process that introduces an error. As the number of discrete codes increases, the corresponding step width gets smaller and the transfer function approaches an ideal straight line. The steps are designed to have transitions such that the midpoint of each step corresponds to the point on this ideal line.

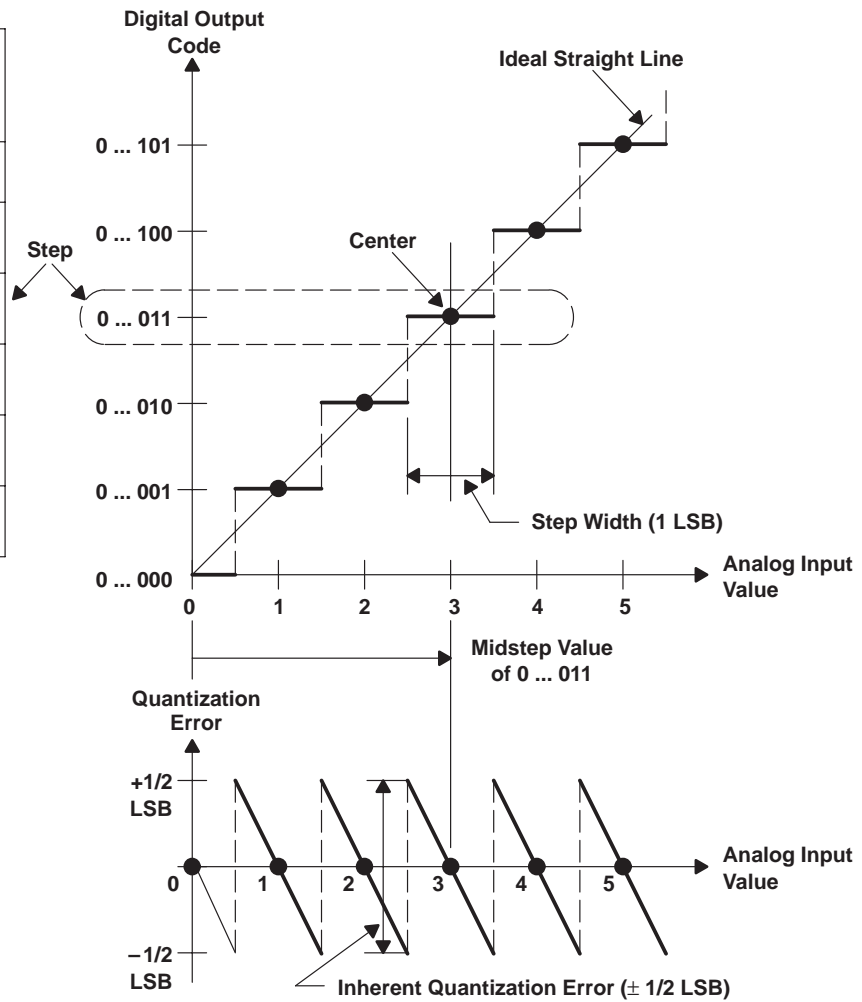
The width of one step is defined as 1 LSB (one least significant bit) and this is often used as the reference unit for other quantities in the specification. It is also a measure of the resolution of the converter since it defines the number of divisions or units of the full analog range. Hence, 1/2 LSB represents an analog quantity equal to one half of the analog resolution.

The resolution of an ADC is usually expressed as the number of bits in its digital output code. For example, an ADC with an n-bit resolution has 2^n possible digital codes which define 2^n step levels. However, since the first (zero) step and the last step are only one half of a full width, the full-scale range (FSR) is divided into $2^n - 1$ step widths.

Hence

$$1 \text{ LSB} = \text{FSR} / (2^n - 1) \text{ for an n-bit converter}$$

CONVERSION CODE	
RANGE OF ANALOG INPUT VALUES	DIGITAL OUTPUT CODE
4.5 • 5.5	0 ... 101
3.5 • 4.5	0 ... 100
2.5 • 3.5	0 ... 011
1.5 • 2.5	0 ... 010
0.5 • 1.5	0 ... 001
0 • 0.5	0 ... 000

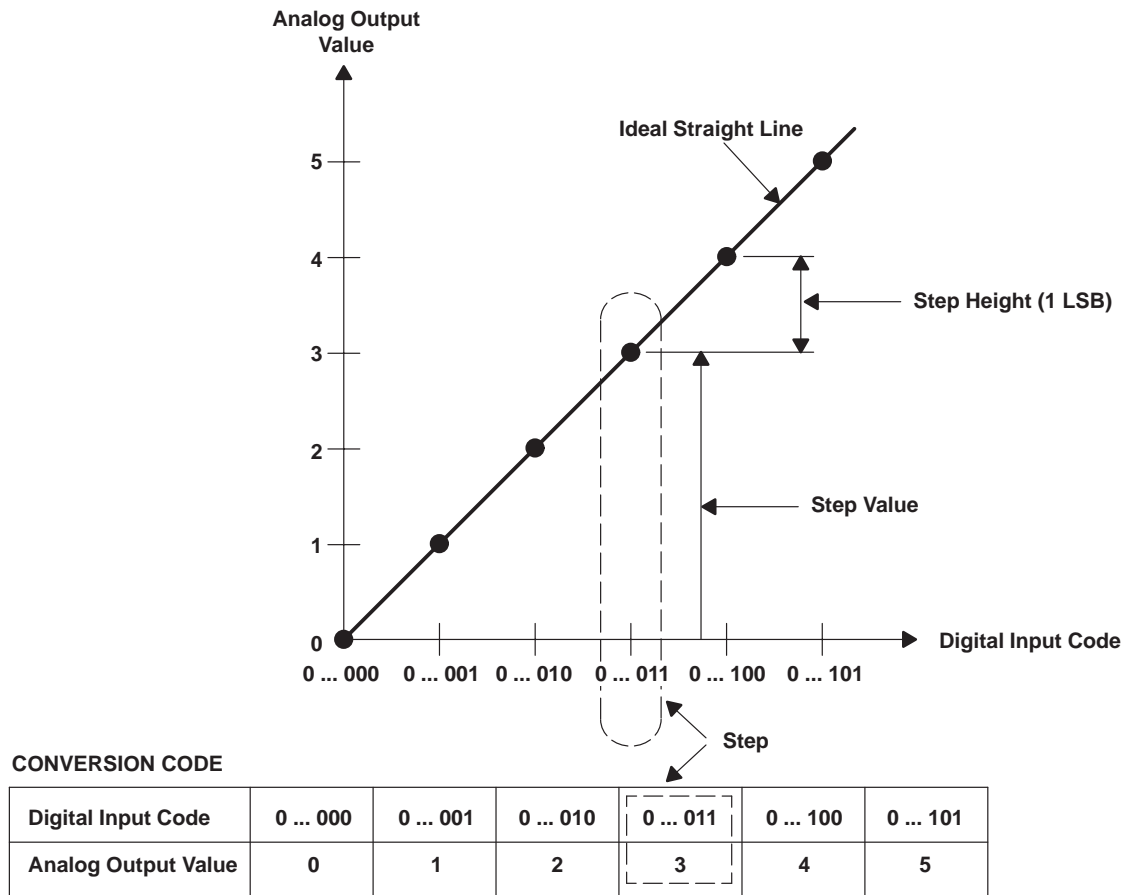


Elements of Transfer Diagram for an Ideal Linear ADC

Figure 1. The Ideal Transfer Function (ADC)

2.2 Digital-to-Analog Converter (DAC)

A DAC represents a limited number of discrete digital input codes by a corresponding number of discrete analog output values. Therefore, the transfer function of a DAC is a series of discrete points as shown in Figure 2. For a DAC, 1 LSB corresponds to the height of a step between successive analog outputs, with the value defined in the same way as for the ADC. A DAC can be thought of as a digitally controlled potentiometer whose output is a fraction of the full scale analog voltage determined by the digital input code.



Elements of Transfer Diagram for an Ideal Linear DAC

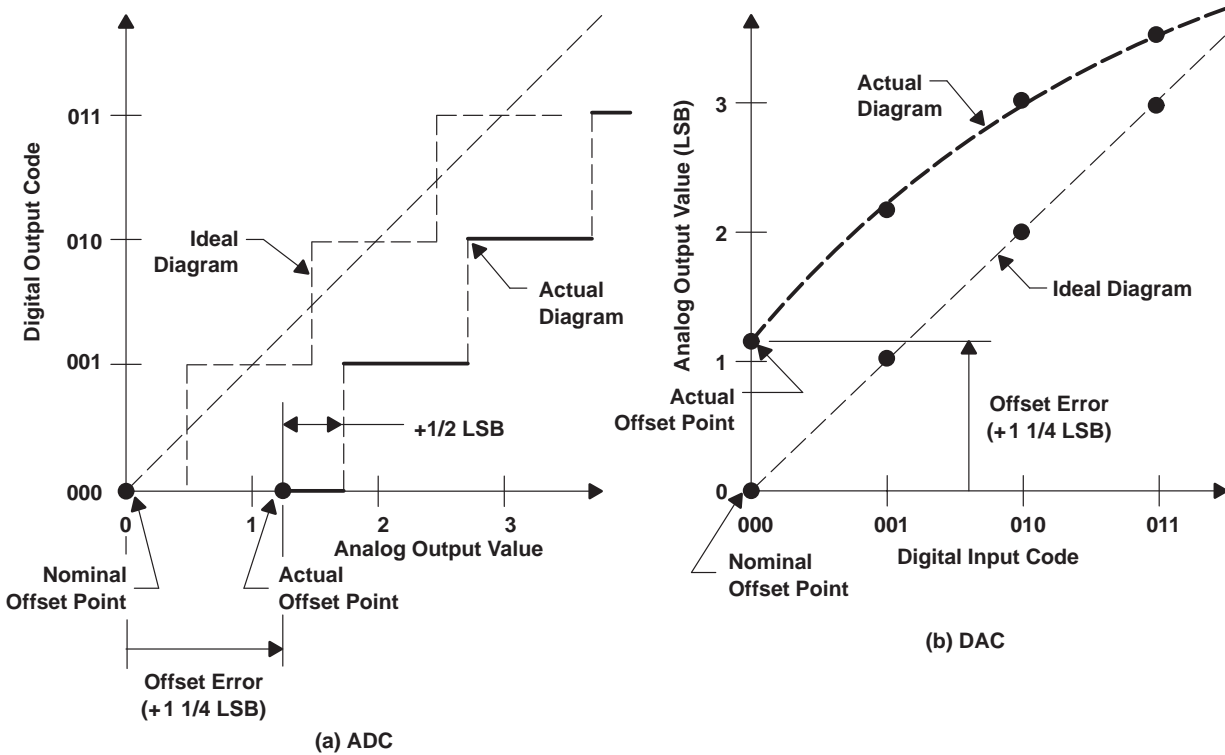
Figure 2. The Ideal Transfer Function (DAC)

3 SOURCES OF STATIC ERROR

Static errors, that is those errors that affect the accuracy of the converter when it is converting static (dc) signals, can be completely described by just four terms. These are offset error, gain error, integral nonlinearity and differential nonlinearity. Each can be expressed in LSB units or sometimes as a percentage of the FSR. For example, an error of 1/2 LSB for an 8-bit converter corresponds to 0.2% FSR.

3.1 Offset Error

The offset error as shown in Figure 3 is defined as the difference between the nominal and actual offset points. For an ADC, the offset point is the midstep value when the digital output is zero, and for a DAC it is the step value when the digital input is zero. This error affects all codes by the same amount and can usually be compensated for by a trimming process. If trimming is not possible, this error is referred to as the zero-scale error.

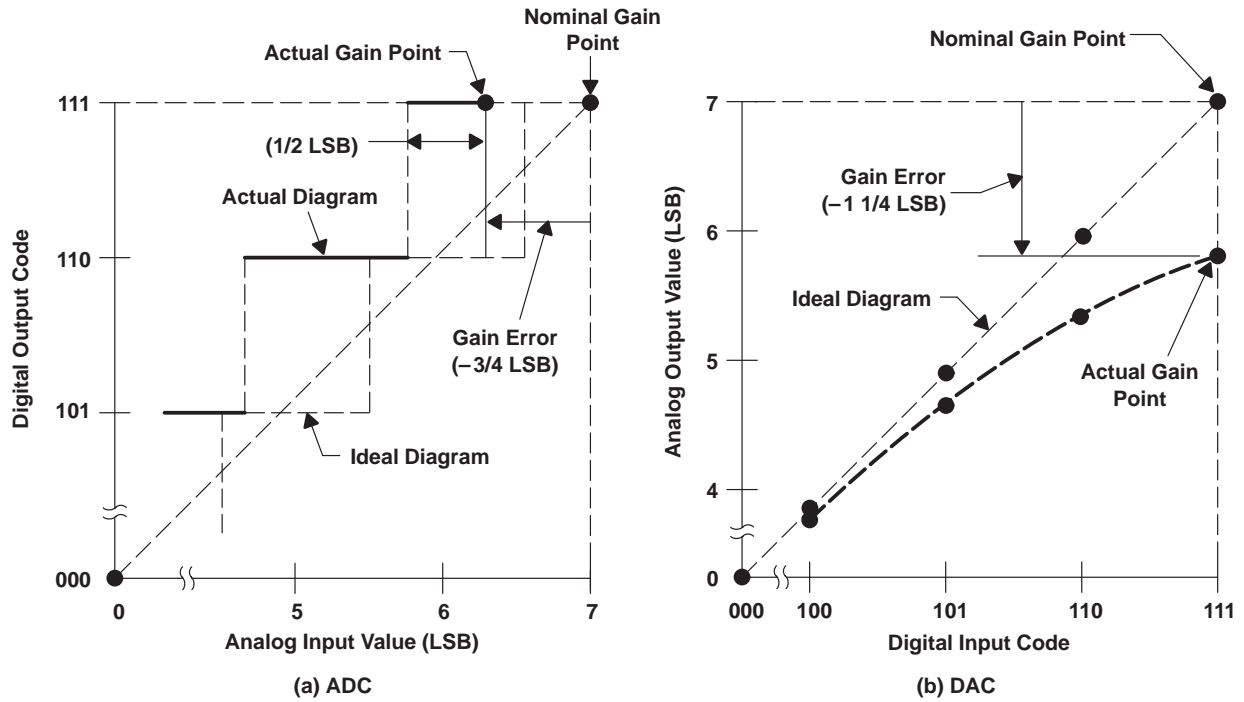


(a) ADC
 Offset error of a Linear 3-Bit Natural Binary Code Converter
 (Specified at Step 000)

Figure 3. Offset Error

3.2 Gain Error

The gain error shown in Figure 4 is defined as the difference between the nominal and actual gain points on the transfer function after the offset error has been corrected to zero. For an ADC, the gain point is the midstep value when the digital output is full scale, and for a DAC it is the step value when the digital input is full scale. This error represents a difference in the slope of the actual and ideal transfer functions and as such corresponds to the same percentage error in each step. This error can also usually be adjusted to zero by trimming.

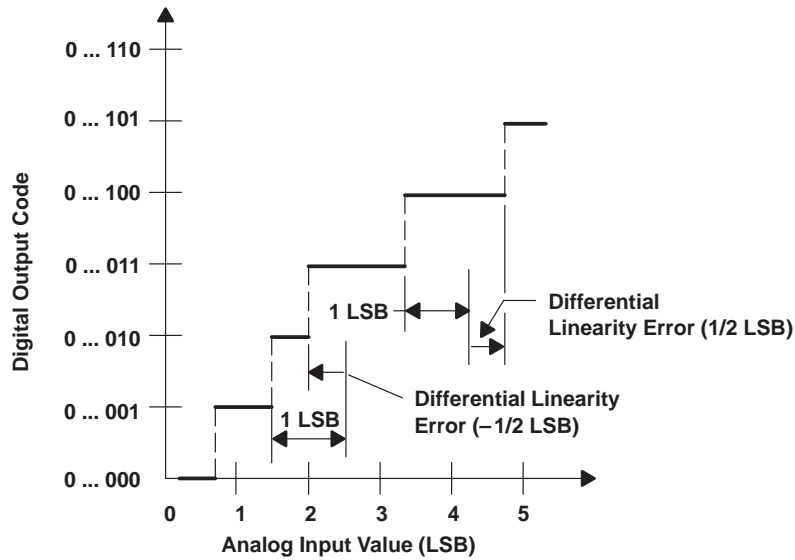


Gain Error of a Linear 3-Bit Natural Binary Code Converter (Specified at Step 111), After Correction of the Offset Error

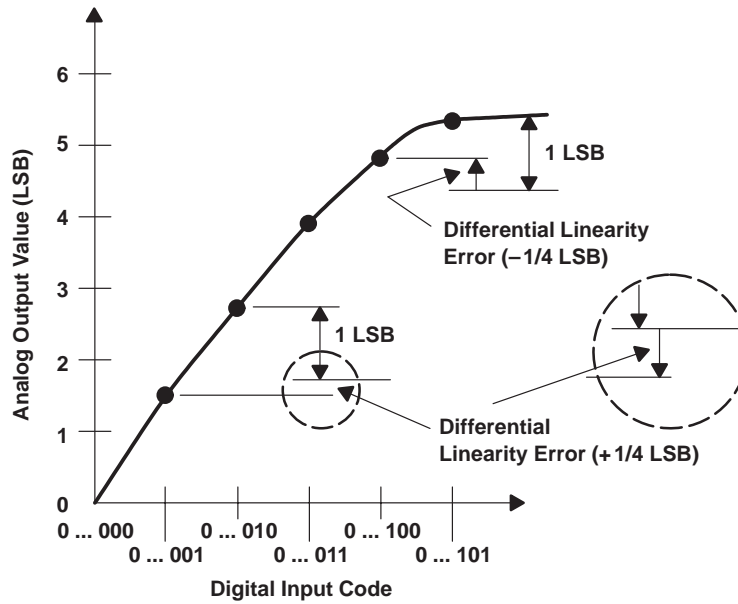
Figure 4. Gain Error

3.3 Differential Nonlinearity (DNL) Error

The differential nonlinearity error shown in Figure 5 (sometimes seen as simply differential linearity) is the difference between an actual step width (for an ADC) or step height (for a DAC) and the ideal value of 1 LSB. Therefore if the step width or height is exactly 1 LSB, then the differential nonlinearity error is zero. If the DNL exceeds 1 LSB, there is a possibility that the converter can become nonmonotonic. This means that the magnitude of the output gets smaller for an increase in the magnitude of the input. In an ADC there is also a possibility that there can be missing codes i.e., one or more of the possible 2^n binary codes are never output.



(a) ADC



(b) DAC

Differential Linearity Error of a Linear ADC or DAC

Figure 5. Differential Nonlinearity (DNL)

3.4 Integral Nonlinearity (INL) Error

The integral nonlinearity error shown in Figure 6 (sometimes seen as simply linearity error) is the deviation of the values on the actual transfer function from a straight line. This straight line can be either a best straight line which is drawn so as to minimize these deviations or it can be a line drawn between the end points of the transfer function once the gain and offset errors have been nullified. The second method is called end-point linearity and is the usual definition adopted since it can be verified more directly.

For an ADC the deviations are measured at the transitions from one step to the next, and for the DAC they are measured at each step. The name integral nonlinearity derives from the fact that the summation of the differential nonlinearities from the bottom up to a particular step, determines the value of the integral nonlinearity at that step.

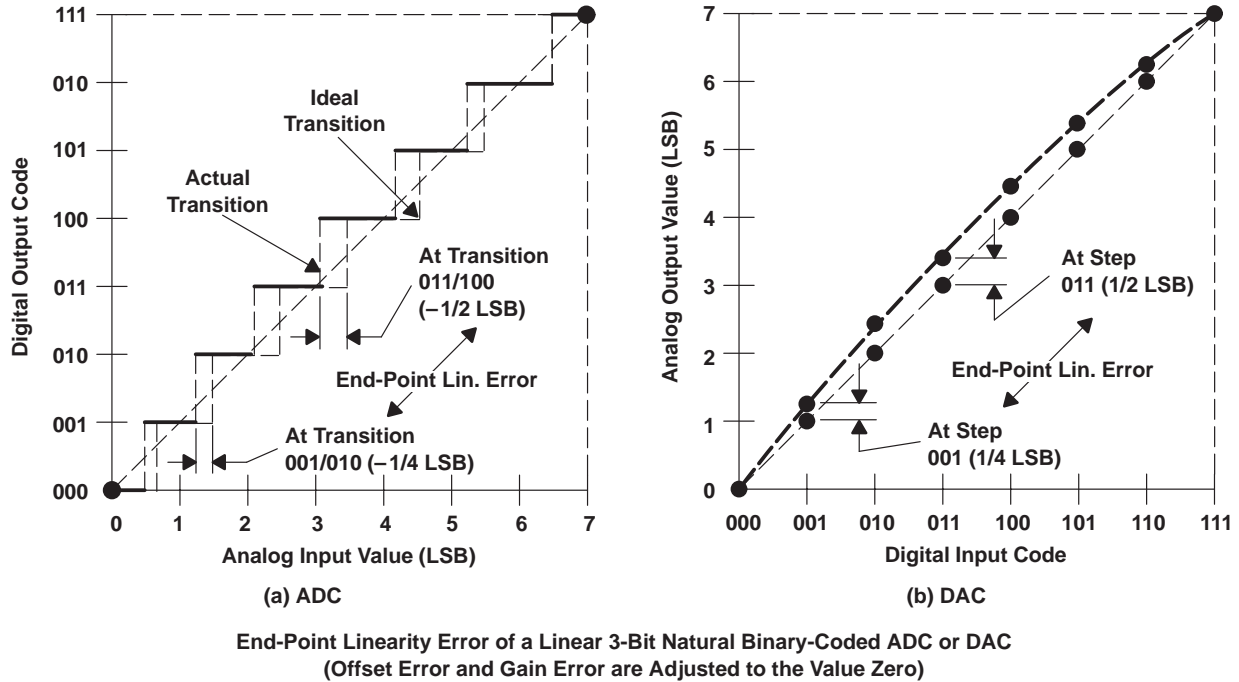
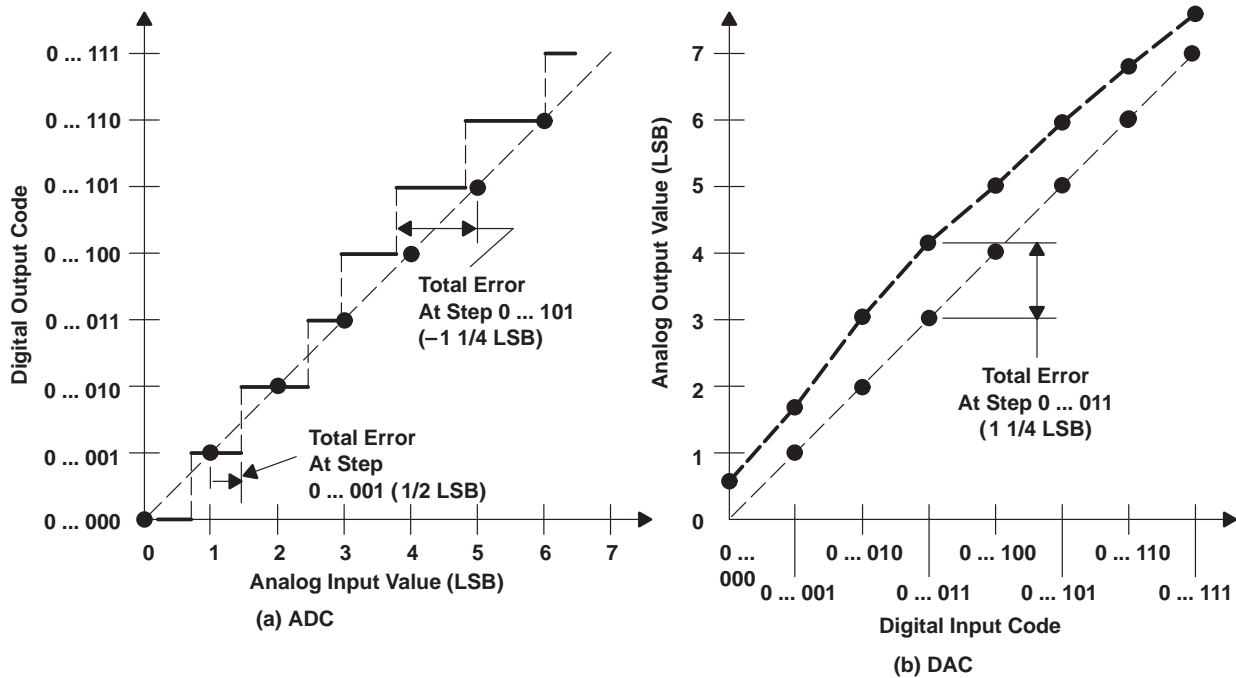


Figure 6. Integral Nonlinearity (INL) Error

3.5 Absolute Accuracy (Total) Error

The absolute accuracy or total error of an ADC as shown in Figure 7 is the maximum value of the difference between an analog value and the ideal midstep value. It includes offset, gain, and integral linearity errors and also the quantization error in the case of an ADC.



Absolute Accuracy or Total Error of a Linear ADC or DAC

Figure 7. Absolute Accuracy (Total) Error

4 APERTURE ERROR

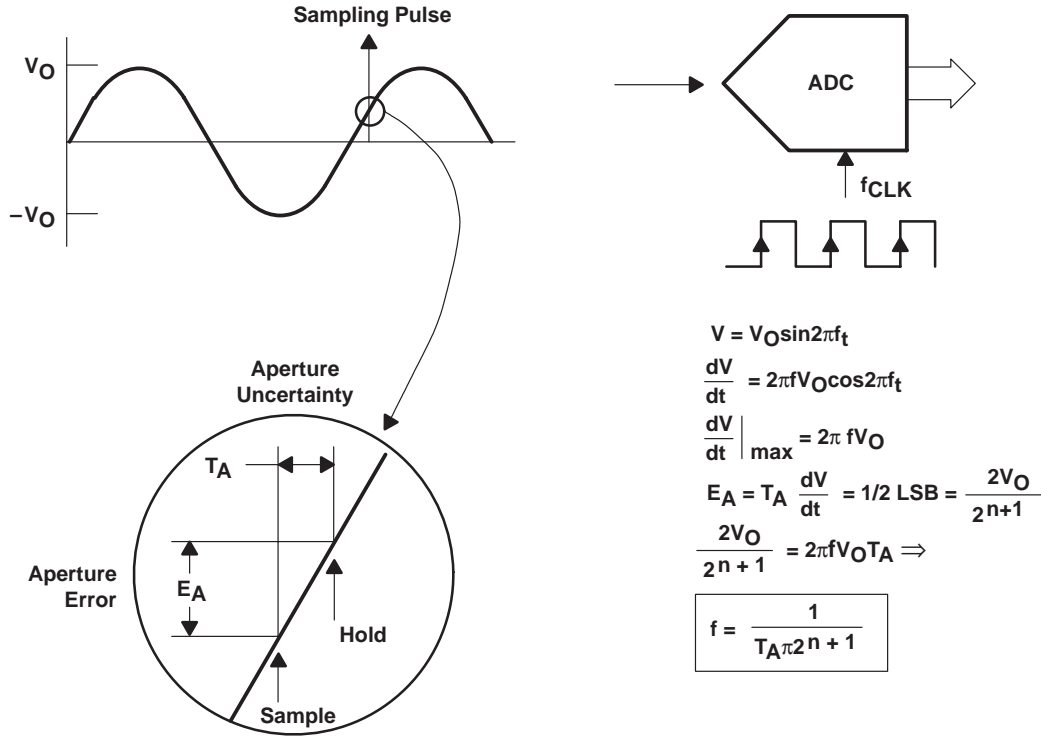


Figure 8. Aperture Error

Aperture error is caused by the uncertainty in the time at which the sample/hold goes from sample mode to hold mode as shown in Figure 8. This variation is caused by noise on the clock or the input signal. The effect of the aperture error is to set another limitation on the maximum frequency of the input sine wave because it defines the maximum slew rate of that signal. For a sine wave input as shown, the value of the input V is defined as:

$$V = V_O \sin 2\pi ft$$

The maximum slew rate occurs at the zero crossing point and is given by:

$$\left. \frac{dV}{dt} \right|_{\max} = 2\pi f V_O$$

If the aperture error is not to affect the accuracy of the converter, it must be less than 1/2 LSB at the point of maximum slew rate. For an n bit converter therefore:

$$E_A = T_A \frac{dV}{dt} = 1/2 \text{ LSB} = \frac{2 V_O}{2^n + 1}$$

Substituting into this gives

$$\frac{2 V_O}{2^n + 1} = 2\pi f V_O T_A$$

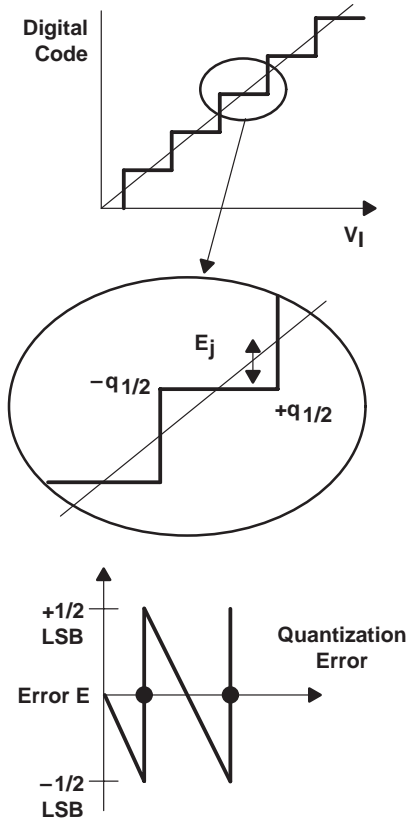
So that the maximum frequency is given by

$$f_{\text{MAX}} = \frac{1}{T_A \pi 2^n + 1}$$

5 QUANTIZATION EFFECTS

The real world analog input to an ADC is a continuous signal with an infinite number of possible states, whereas the digital output is by its nature a discrete function with a number of different states determined by the resolution of the device. It follows from this therefore, that in converting from one form to the other, certain parts of the analog signal that were represented by a different voltage on the input are represented by the same digital code at the output. Some information has been lost and distortion has been introduced into the signal. This is quantization noise.

For the ideal staircase transfer function of an ADC, the error between the actual input and its digital form has a uniform probability density function if the input signal is assumed to be random. It can vary in the range $\pm 1/2$ LSB or $\pm q/2$ where q is the width of one step as shown in Figure 9.



Error at the j th step

$$E_j = (V_j - V_i)$$

The mean square error over the step

$$\overline{E_j^2} = \frac{1}{q} \int_{-q/2}^{+q/2} E_j^2 dE = \frac{q^2}{12}$$

Assuming equal steps, the total error is

$$\overline{N^2} = q^2/12 \text{ (Mean square quantization noise)}$$

For an input sine wave $F(t) = A \sin \omega t$, the signal power

$$\overline{F^2(t)} = \frac{1}{2\pi} \int_0^{2\pi} A^2 \sin^2 \omega t d\omega t = \frac{A^2}{2}$$

$$\text{and } q = \frac{2A}{2^n} = \frac{A}{2^{n-1}}$$

$$\text{SNR} = 10 \text{ Log} \left(\frac{F^2}{N^2} \right) = 10 \text{ Log} \left(\frac{A^2/2}{A^2/3 \times 2^n} \right)$$

$$\text{SNR} = 6.02n + 1.76 \text{ dB}$$

Figure 9. Quantization Effects

Where

$$p(\epsilon) = \frac{1}{q} \text{ for } \left(-\frac{q}{2} \leq \epsilon \leq +\frac{q}{2} \right)$$

Otherwise

$$p(\epsilon) = 0$$

The average noise power (mean square) of the error over a step is given by

$$\overline{N^2} = \frac{1}{q} \int_{-q/2}^{q/2} \epsilon^2 d\epsilon$$

which gives

$$\overline{N^2} = \frac{q^2}{12}$$

The total mean square error, N^2 , over the whole conversion area is the sum of each quantization levels mean square multiplied by its associated probability. Assuming the converter is ideal, the width of each code step is identical and therefore has an equal probability. Hence for the ideal case

$$N^2 = \frac{q^2}{12}$$

Considering a sine wave input $F(t)$ of amplitude A so that

$$F(t) = A \sin \omega t$$

which has a mean square value of $F^2(t)$, where

$$F^2(t) = \frac{1}{2\pi} \int_0^{2\pi} A^2 \sin^2(\omega t) dt$$

which is the signal power. Therefore the signal to noise ratio SNR is given by

$$\text{SNR(dB)} = 10 \text{Log} \left[\left(\frac{A^2}{2} \right) / \left(\frac{q^2}{12} \right) \right]$$

But

$$q = 1 \text{ LSB} = \frac{2A}{2^n} = \frac{A}{2^{n-1}}$$

Substituting for q gives

$$\begin{aligned} \text{SNR(dB)} &= 10 \text{Log} \left[\left(\frac{A^2}{2} \right) / \left(\frac{A^2}{3 \times 2^{2n}} \right) \right] = 10 \text{Log} \left(\frac{3 \times 2^{2n}}{2} \right) \\ &\Rightarrow \underline{\underline{6.02n + 1.76 \text{dB}}} \end{aligned}$$

This gives the ideal value for an n bit converter and shows that each extra 1 bit of resolution provides approximately 6 dB improvement in the SNR.

In practice, the errors mentioned in section 3 introduce nonlinearities that lead to a reduction of this value. The limit of a $1/2$ LSB differential linearity error is a missing code condition which is equivalent to a reduction of 1 bit of resolution and hence a reduction of 6 dB in the SNR. This then gives a worst case value of SNR for an n -bit converter with $1/2$ LSB linearity error.

$$\text{SNR (worst case)} = 6.02n + 1.76 - 6 = 6.02n - 4.24 \text{ dB}$$

Hence we have established the boundary conditions for the choice of the resolution of the converter based upon a desired level of SNR.

6 IDEAL SAMPLING

In converting a continuous time signal into a discrete digital representation, the process of sampling is a fundamental requirement. In an ideal case, sampling takes the form of a pulse train of impulses which are infinitesimally narrow yet have unit area. The reciprocal of the time between each impulse is called the sampling rate. The input signal is also idealized by being truly bandlimited, containing no components in its spectrum above a certain value (see Figure 10).

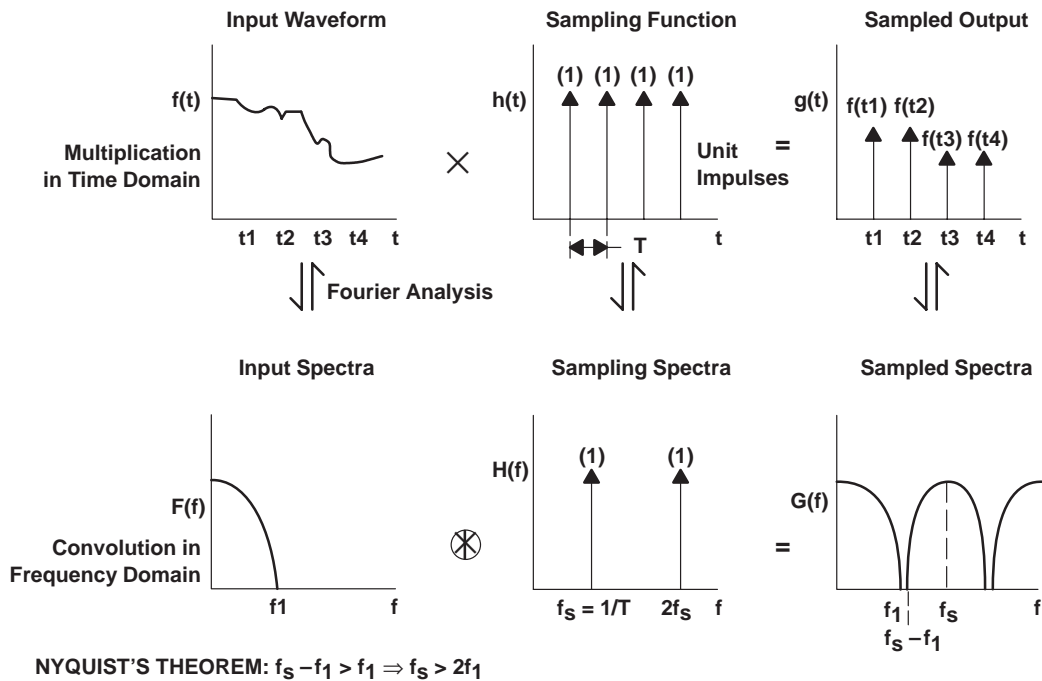


Figure 10. Ideal Sampling

The ideal sampling condition shown is represented in both the frequency and time domains. The effect of sampling in the time domain is to produce an amplitude modulated train of impulses representing the value of the input signal at the instant of sampling. In the frequency domain, the spectrum of the pulse train is a series of discrete frequencies at multiples of the sampling rate. Sampling convolves the spectra of the input signal with that of the pulse train to produce the combined spectrum shown, with double sidebands around each discrete frequency which are produced by the amplitude modulation. In effect some of the higher frequencies are folded back so that they produce interference at lower frequencies. This interference causes distortion which is called aliasing.

If the input signal is bandlimited to a frequency f_1 and is sampled at frequency f_s , as shown in the figure, overlap (and hence aliasing) does not occur if

$$f_1 < f_s - f_1 \quad \text{i.e., } 2f_1 < f_s$$

Therefore if sampling is performed at a frequency at least twice as great as the maximum frequency of the input signal, no aliasing occurs and all of the signal information can be extracted. This is Nyquist's Sampling Theorem, and it provides the basic criteria for the selection of the sampling rate required by the converter to process an input signal of a given bandwidth.

7 REAL SAMPLING

The concept of an impulse is a useful one to simplify the analysis of sampling. However, it is a theoretical ideal which can be approached but never reached in practice. Instead the real signal is a series of pulses with the period equalling the reciprocal of the sampling frequency. The result of sampling with this pulse train is a series of amplitude modulated pulses (see Figure 11).

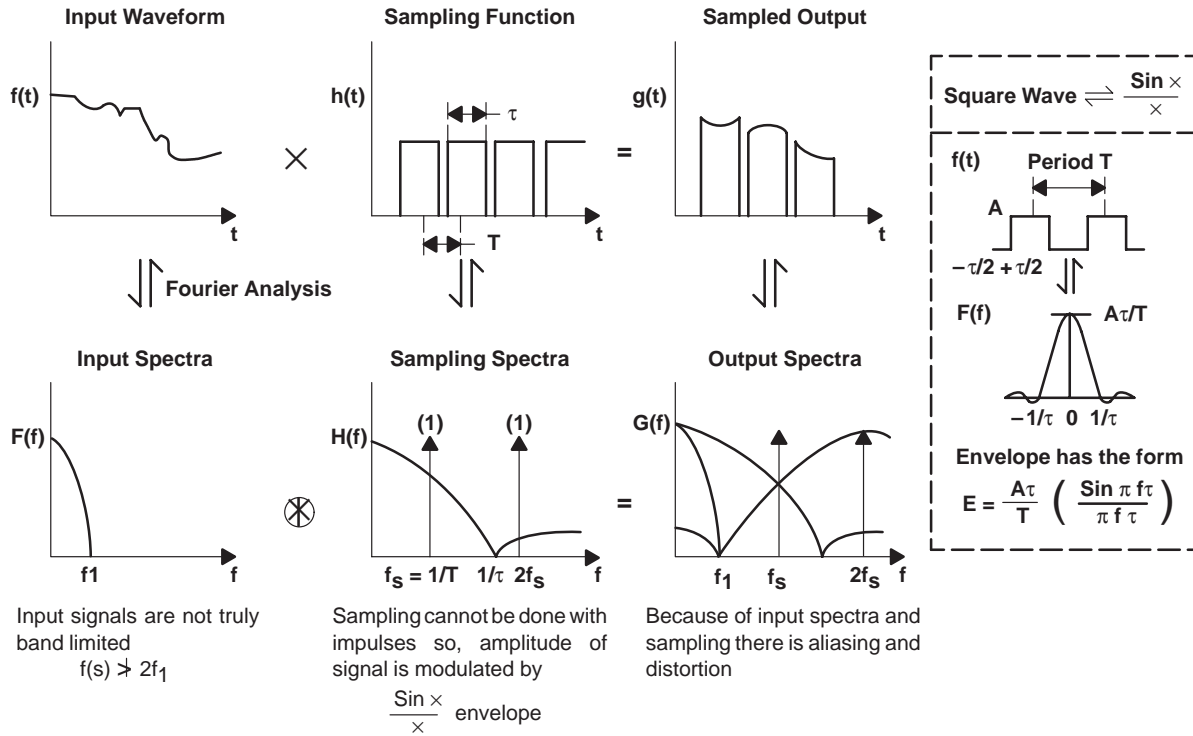


Figure 11. Real Sampling

Examining the spectrum of the square wave pulse train shows a series of discrete frequencies, as with the impulse train, but the amplitude of these frequencies is modified by an envelope which is defined by $(\sin x)/x$ [sometimes written $\text{sinc}(x)$] where x in this case is $\pi f_s \tau$. For a square wave of amplitude A , the envelope of the spectrum is defined as

$$\text{Envelope} = A \left(\frac{\tau}{T} \right) \left[\frac{\sin(\pi f_s \tau)}{\pi f_s \tau} \right]$$

The error resulting from this can be controlled with a filter which compensates for the sinc envelope. This can be implemented as a digital filter, in a DSP, or using conventional analog techniques.

8 ALIASING EFFECTS AND CONSIDERATIONS

No signal is truly deterministic and therefore in practice has infinite bandwidth. However, the energy of higher frequency components becomes increasingly smaller so that at a certain value it can be considered to be irrelevant. This value is a choice that must be made by the system designer.

As shown, the amount of aliasing is affected by the sampling frequency and by the relevant bandwidth of the input signal, filtered as required. The factor that determines how much aliasing can be tolerated is ultimately the resolution of the system. If the system has low resolution, then the noise floor is already relatively high and aliasing does not have a significant effect. However, with a high resolution system, aliasing can increase the noise floor considerably and therefore needs to be controlled more completely.

One way to prevent aliasing is to increase the sampling rate, as shown. However, the frequency is limited by the type of converter used and also by the maximum clock rate of the digital processor receiving and transmitting the data. Therefore, to reduce the effects of aliasing to within acceptable levels, analog filters must be used to alter the input signal spectrum (see Figure 12).

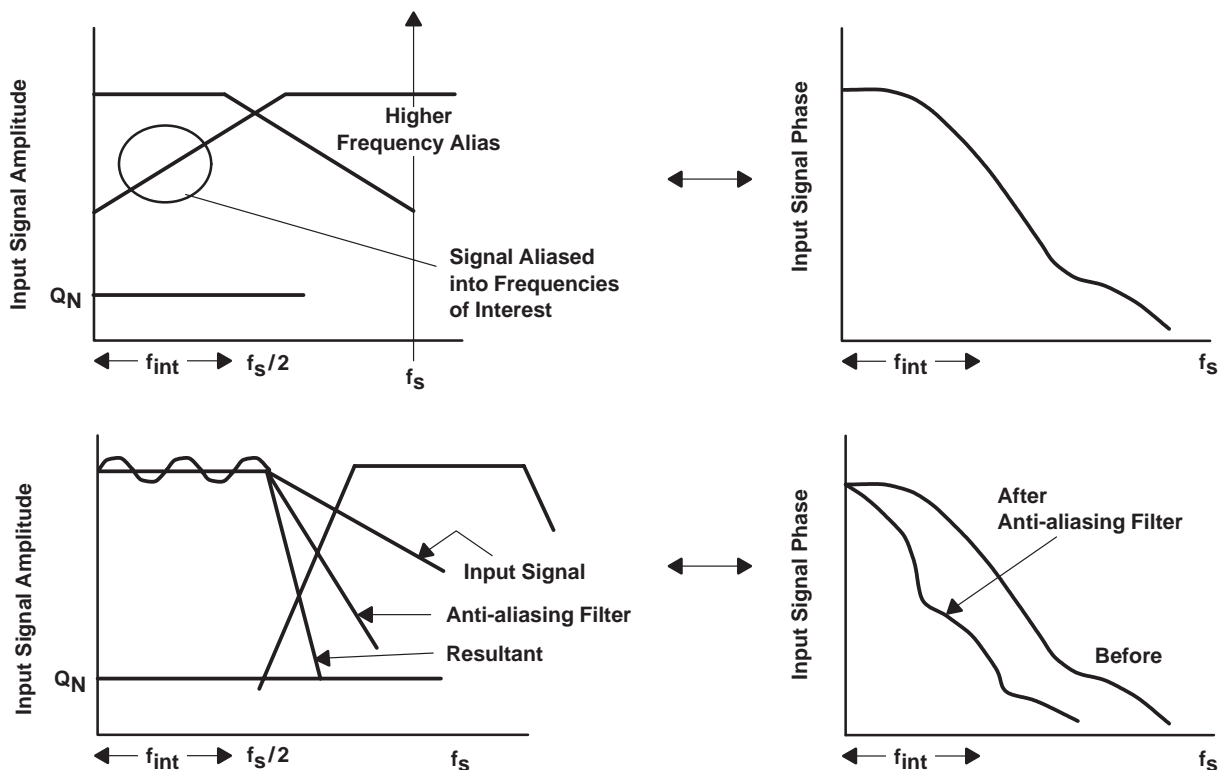


Figure 12. Aliasing Effects and Considerations

8.1 Choice of Filter

As shown with sampling, there is an ideal solution to the choice of a filter and a practical realization that compromises must be made. The ideal filter is a so-called brickwall filter which introduces no attenuation in the passband, and then cuts down instantly to infinite attenuation in the stopband. In practice, this is approximated by a filter that introduces some attenuation in the passband, has a finite rolloff, and passes some frequencies in the stopband. It can also introduce phase distortion as well as amplitude distortion. The choice of the filter order and type must be decided upon so as to best meet the requirements of the system.

8.2 Types of Filter

The basic types of filters available to the designer are briefly presented for comparison purposes. This is not intended to be a full analysis of the subject; therefore, other texts should be referenced for more details.

8.2.1 Butterworth Filter

A Butterworth (maximally flat) filter is the most commonly used general purpose filter. It has a monotonic passband with the attenuation increasing up to its 3-dB point which is known as the natural frequency. This frequency is the same regardless of the order of the filter. However, by increasing the order of the filter, the roll-off in the passband moves closer to its natural frequency and the roll-off in the transition region between the natural frequency and the stopband becomes sharper.

8.2.2 Chebyshev Filter

The Chebyshev equal ripple filter distributes the roll-off across the whole passband. It introduces more ripple in the passband but provides a sharper roll-off in the transition region. This type of filter has poorer transient and step responses due to its higher Q values in the stages of the filter.

8.2.3 Inverse Chebyshev Filter

Both the Butterworth and Chebyshev filters are monotonic in the transition region and stopband. Since ripple is allowed in the stopband, it is possible to make the roll-off sharper. This is the principle of the Inverse Chebyshev, based on the reciprocal of the angular frequency in the Chebyshev filter response. This filter is monotonic in the passband and can be flatter than the Butterworth filter while providing a greater initial roll-off than the Chebyshev filter.

8.2.4 Cauer Filter

The Cauer or (Elliptic) filter is nonmonotonic in both the pass and stop bands, but provides the greatest roll-off in any of the standard filter configurations.

8.2.5 Bessel-Thomson Filter

All of the types mentioned above introduce nonlinearities into the phase relationship of the component frequencies of the input spectrum. This can be a problem in some applications when the signal is reconstructed. The Bessel-Thomson or linear delay filter is designed to introduce no phase distortion but this is achieved at the expense of a poorer amplitude response.

In general, the performance of all of these types can be improved by increasing the number of stages, i.e., the order of the filter. The penalty for this of course is the increased cost of components and board space required. For this reason, an integrated solution using switched capacitor filter building blocks which provide comparable performance with a discrete solution over a range of frequencies from about 1 kHz to 100 kHz might be appropriate. They also provide the designer with a compact and cost effective solution.

8.3 TLC04 Anti-Aliasing Butterworth Filter

The TLC04 fourth order Butterworth filter features include the following:

- Low clock to cutoff frequency error . . . 0.8%
- Cutoff depends only on stability of external clock
- Cutoff range of 0.1 Hz to 30 kHz
- 5-V to 12-V operation
- Self clocking or both TTL and COS compatible

As detailed previously the Butterworth filter generally provides the best compromise in filter configurations and is by far the easiest to design. The Butterworth filter's characteristic is based on a circle which means that when designing filters, all stages to the filter have the same natural frequency enabling simpler filter design. Most modern designs which use operational amplifiers are based on building the whole transfer function by a series of second order numerator and denominator stages (a Biquad stage). The Butterworth design is simplified, when using these stages, because each stage has the same natural frequency. This can easily be converted to a switched capacitor filter (SCF) which has very good capacitor matching and accurately synthesized RC time constants.

The switched capacitor technique is demonstrated in Figure 13. Two clocks operating at the same frequency but in complete antiphase, alternately connect the capacitor C_2 to the input and the inverting input of an operational amplifier. During Φ_1 , charge Q flows onto the capacitor equal to $V_1 C_2$. The switch is considered to be ideal so that there is no series resistance and the capacitor charges instantaneously. During Φ_2 , the switches change so that C_2 is now connected to the virtual earth at the operational amplifier input. It discharges instantaneously delivering the stored charge Q .

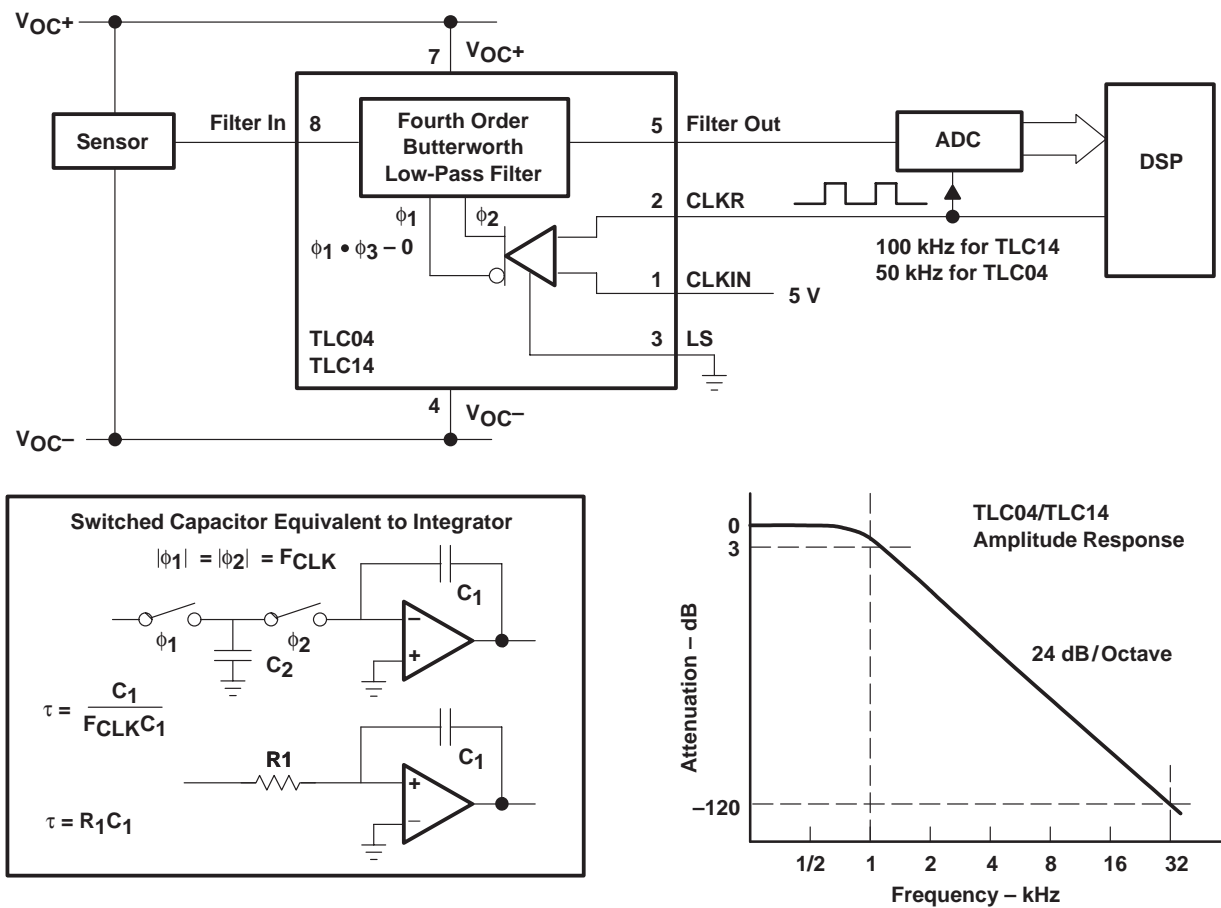


Figure 13. TLC04 Anti-aliasing Butterworth Filter

The average current that flows I_{AV} depends on the frequency of the clocks T so that

$$I_{AV} = \frac{Q}{T} = V_I \frac{C_2}{T} = V_I C_2 F_{CLK}$$

Therefore, the switched capacitor looks like a resistor of value

$$R_{eq} = \frac{V_I}{I_{AV}} = \frac{1}{C_2 F_{CLK}}$$

The advantage of the technique is that the time constant of the integrator can be programmed by altering this equivalent resistance, and this is done by simply altering the clock frequency. This provides precision in the filter design, because the time constant then depends on the ratio of two capacitors which can be fabricated in silicon to track each other very closely with voltage and temperature. Note that the analysis assumes V_I to be constant so that for an ac signal, the clock frequency must be much higher than the frequency of the input.

The TLC04 is one such filter which is internally configured to provide the Butterworth low-pass filter response, and the cut-off frequency for the device is controlled by a digital clock. For this device, the cut-off frequency is set simply by the clock frequency so that the clock to cut-off frequency ratio is 50:1 with an accuracy of 0.8%. This enables the cut-off frequency of the filter to be tied to the sampling rate, so that only one fundamental clock signal is required for the system as a whole. Another advantage of SCF techniques means that fourth order filters can be attained using only one integrated circuit and they are much more easily controlled.

The response of an n th order Butterworth filter is described by the following equation.

$$\text{Attenuation} = \left[1 + \left(\frac{f}{f_c} \right)^{2n} \right]^{1/2}$$

The table below lists the fourth order realization in the TLC04.

FREQUENCY	ATTENUATION (FACTOR)	ATTENUATION (dB)	PHASE (DEGREE)
$F_c/2$	0.998	0.02	26.6
F_c	0.707	3	45
$2F_c$	0.0624	24	63.4
$4F_c$	0.00391	48	76
$8F_c$	0.000244	72	82.9
$12F_c$	0.000048	86	85.2
$16F_c$	0.000015	96	86.4

This means that sampling at 8 times the cut off frequency gives an input-to-aliased signal ratio of 72 dB, which is less than ten bit quantization noise distortion.

