
The Importance of Data Converter Static Specifications— Don't Lose Sight of the Basics!

by Walt Kester

INTRODUCTION

In the 1950s and 1960s, dc performance specifications such as integral nonlinearity, differential nonlinearity, monotonicity, no missing codes, gain error, offset error, and drift, etc., were primarily used to characterize data converters. These specifications were adequate during this era, because most early applications (with the exception of PCM and radar, for example) dealt with dc or low frequency signals such as those encountered in industrial measurement and process control. With the advent of microprocessors and digital signal processing (DSP) in the 1970s and 1980s, dynamic performance specifications, such as signal-to-noise ratio (SNR), spurious free dynamic range (SFDR), etc., were required in order to adequately characterize converters for more sophisticated signal processing applications.

Modern data converter applications cover the frequency spectrum from low frequency industrial measurement to wideband radio receivers. Even though the importance of dc specifications generally decreases with increasing signal frequency, they are still important in many applications. For instance, significant gain and/or offset errors in IF sampling applications can cause signal clipping and thereby degrade SNR and SFDR. In applications requiring matched converters, such as interleaving, simultaneous sampling, and I/Q signal processing, the relative gain and offset matching between individual converters is critical.

This article explains the dc performance specifications of data converters so that you will be equipped to understand this important part of an ADC or DAC data sheet.

DATA CONVERTER RESOLUTION AND QUANTIZATION

It is important to remember that for both DACs and ADCs, either the input or output is digital, and therefore the signal is 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. As previously discussed, the analog signals will generally be voltages or currents.

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 1.

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 1: Quantization: The Size of a Least Significant Bit (LSB)

TRANSFER FUNCTIONS OF IDEAL DATA CONVERTERS

Figure 2 shows the ideal transfer characteristics for a 3-bit unipolar DAC and 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.

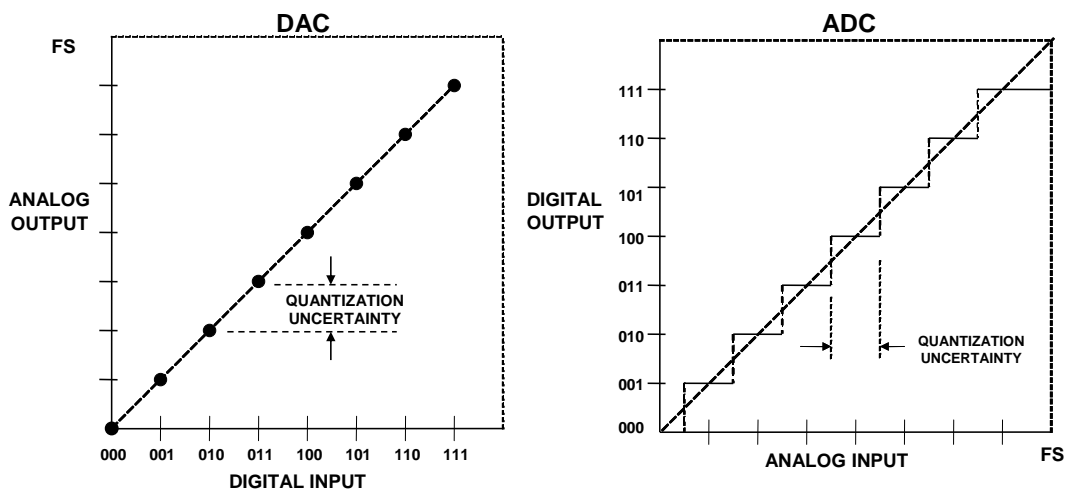


Figure 2: Transfer Functions for Ideal 3-Bit DAC and ADC

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—often referred to as the *code centers*.

For both DACs and ADCs, digital full-scale (all "1"s) corresponds to 1 LSB below the analog full-scale (FS). The (ideal) ADC transitions take place at $\frac{1}{2}$ LSB above zero, and thereafter every LSB, until $1\frac{1}{2}$ 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 $\frac{1}{2}$ LSB between the actual analog input and the exact value of the digital output. This is known as the *quantization error* or *quantization uncertainty* also shown in Figure 2. In ac (sampling) applications this quantization error gives rise to *quantization noise* which is discussed in detail in other articles.

There are many possible digital coding schemes for data converters: *straight binary*, *offset binary*, *1's complement*, *2's complement*, *sign magnitude*, *gray code*, *BCD* and others. This article, 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. For more details on data converter coding, refer to References 1 and 2.

The examples in Figure 2 use *unipolar* converters, whose analog port has only a single polarity. These are the simplest type, but *bipolar* converters are generally more useful in real-world applications. There are two types of bipolar converters: 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). The unipolar, offset binary, and sign-magnitude representations are shown in Figure 3.

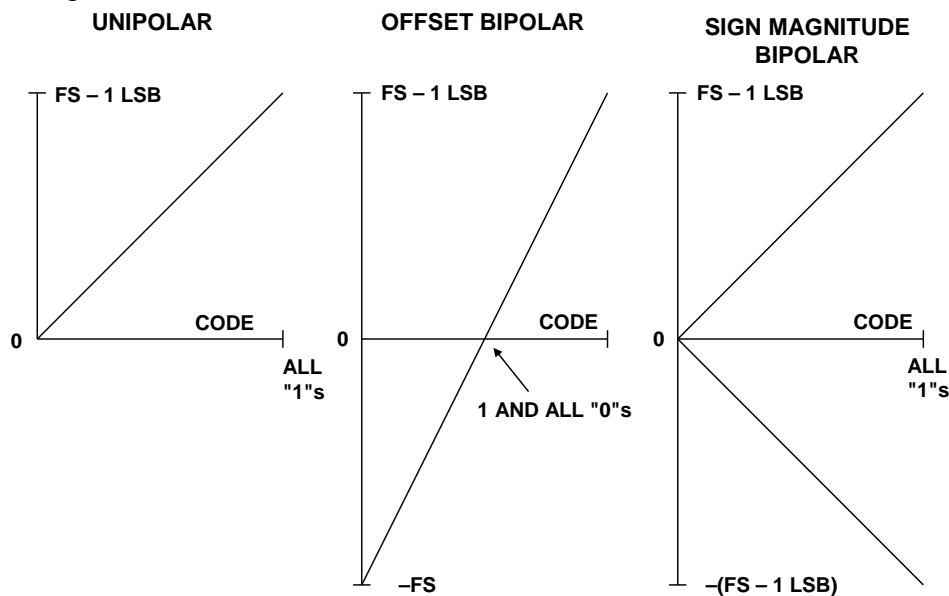


Figure 3: Unipolar and Bipolar Converters

DATA CONVERTER GAIN AND OFFSET ERRORS

The four dc errors in a data converter are *offset error*, *gain error*, and two types of *linearity error* (*differential and integral*). Offset and gain errors are analogous to offset and gain errors in amplifiers as shown in Figure 4 for a bipolar input range. (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.)

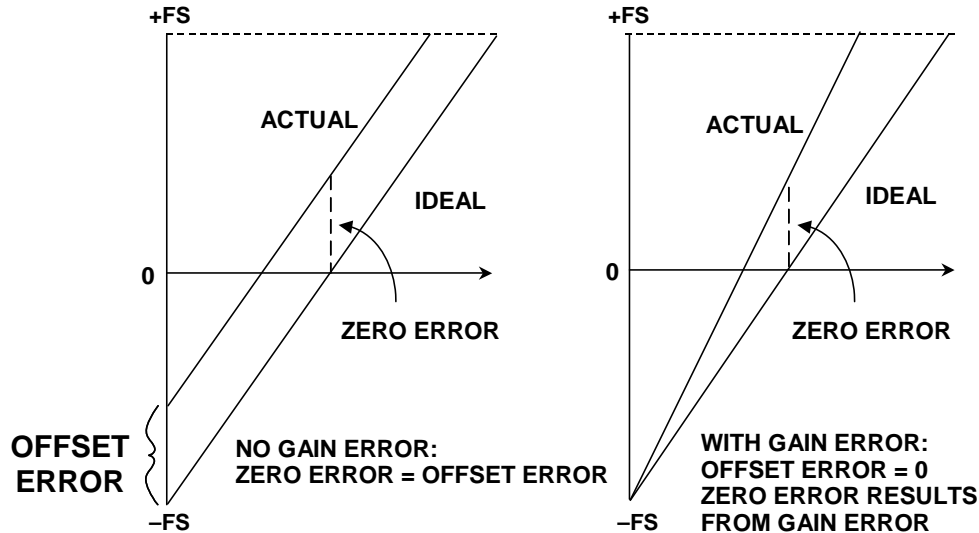


Figure 4: Bipolar Data Converter Offset and Gain Error

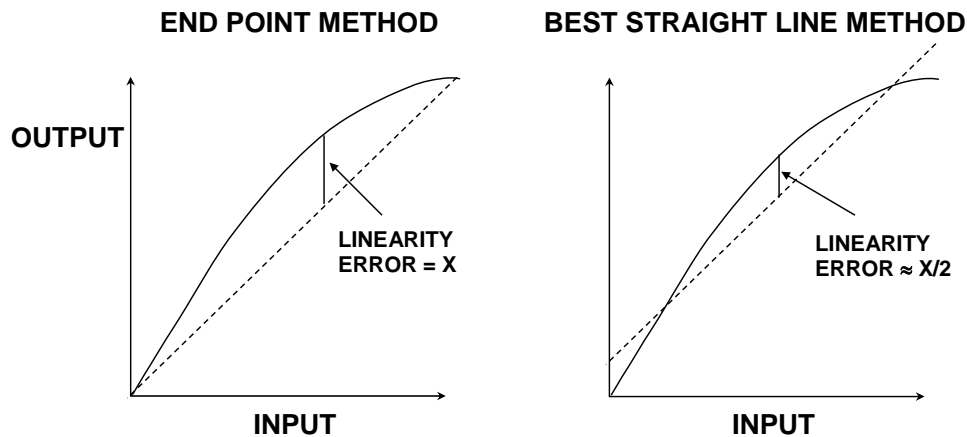
The transfer characteristics of both DACs and ADCs may be expressed as a straight line given by $D = K + GA$, where D is the digital code, A is the analog signal, and K and G are constants. In a unipolar converter, the ideal value of K is zero; in an offset bipolar converter it is -1 MSB. The offset 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.

DATA CONVERTER LINEARITY ERRORS

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). For an ADC, the most popular convention is to draw the straight line through the mid-

points of the codes, or the code centers. There are two common ways of choosing the straight line: *end point* and *best straight line* as shown in Figure 5.



**Figure 5: Method of Measuring Integral Linearity Errors
(Same Converter on Both Graphs)**

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 better to specify distortion than dc linearity, so it is rarely necessary to use the best straight line method to define converter linearity.

The other type of converter nonlinearity is *differential nonlinearity* (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. Differential linearity error is defined as the maximum amount of deviation of any quantum (or LSB change) in the entire transfer function from its ideal size of 1 LSB.

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 across the range of the converter. Figure 6 shows the non-ideal transfer functions for a DAC and an ADC and shows the effects of the DNL error.

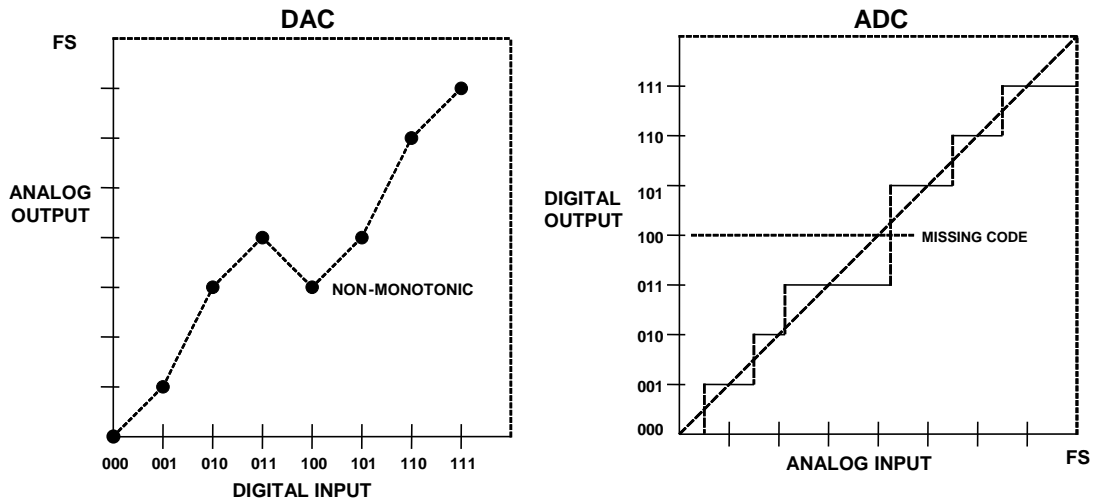


Figure 6: Transfer Functions for Non-Ideal 3-Bit DAC and ADC

The DNL of a DAC is examined more closely in Figure 7. If the DNL of a DAC is less than -1 LSB at any transition, the DAC is *non-monotonic* i.e., its transfer characteristic contains one or more localized maxima or minima. A DNL greater than $+1$ LSB 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., $|DNL| \leq 1$ LSB) then the device must be monotonic, even without an explicit guarantee.

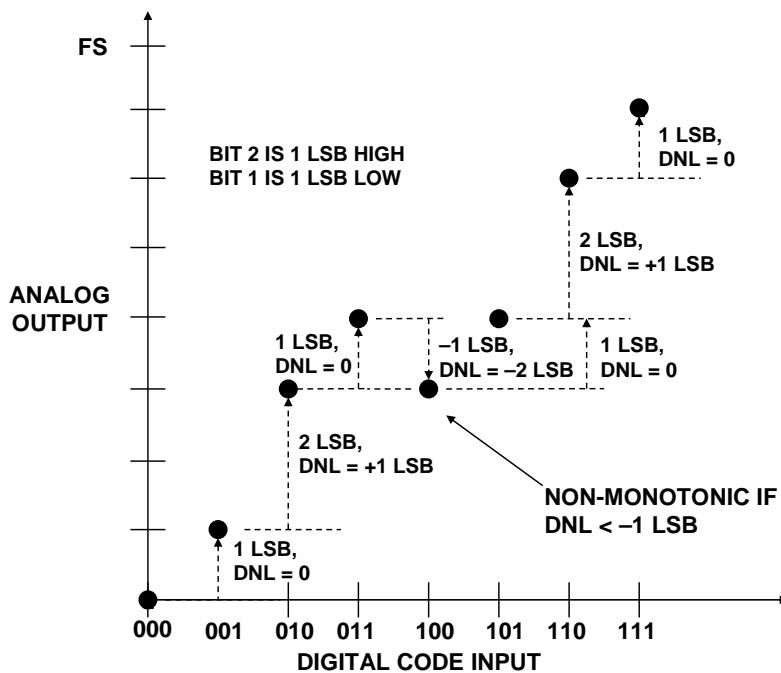


Figure 7: Details of DAC Differential Nonlinearity

In Figure 8, the DNL of an ADC is examined more closely on an expanded scale. ADCs can be non-monotonic, but a more common result of excess DNL in ADCs is *missing codes*. Missing codes in an ADC are as objectionable as non-monotonicity in a DAC. Again, they result from $DNL < -1 \text{ LSB}$.

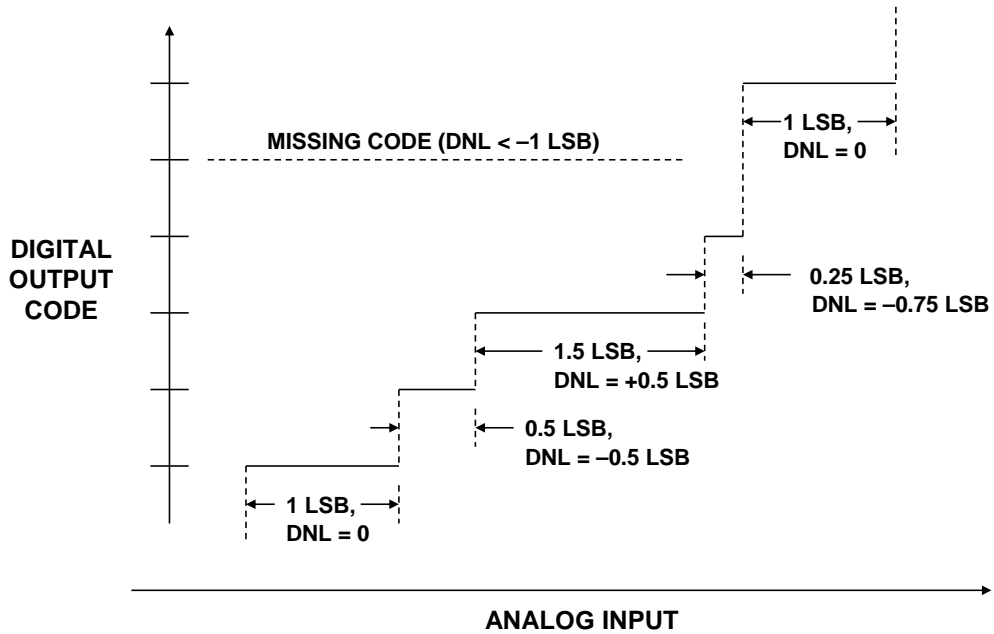


Figure 8: Details of ADC Differential Nonlinearity

Not only can ADCs have missing codes, they can also be non-monotonic as shown in Figure 9. As in the case of DACs, this can present major problems—especially in servo applications.

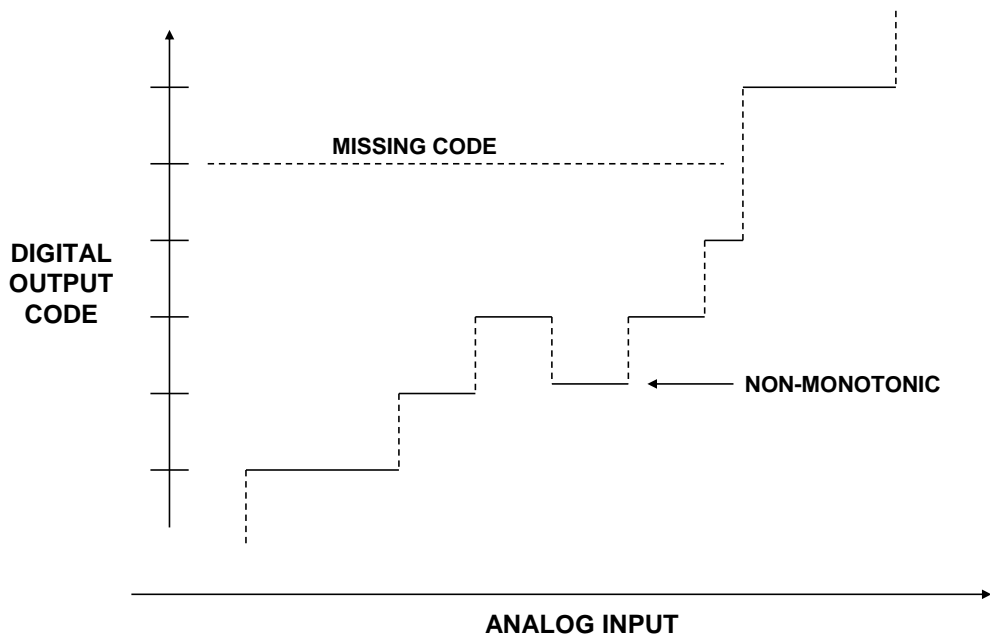


Figure 9: Non-Monotonic ADC with Missing Code

In a DAC, there can be no missing codes—each digital input word will produce a corresponding analog output. However, DACs can be non-monotonic as previously discussed. In a straight binary DAC, the most likely place a non-monotonic condition can develop is at mid-scale between the two codes: 011...11 and 100...00. If a non-monotonic condition occurs here, it is generally because the DAC is not properly calibrated or trimmed. A successive approximation ADC with an internal non-monotonic DAC will generally produce missing codes but remain monotonic. However it is possible for an ADC to be non-monotonic—again depending on the particular conversion architecture. Figure 9 shows the transfer function of an ADC which is non-monotonic and has a missing code.

ADCs which use the *subranging* architecture divide the input range into a number of coarse segments, and each coarse segment is further divided into smaller segments—and ultimately the final code is derived. This process is described in more detail in Chapter 4 of this book. An improperly trimmed subranging ADC may exhibit non-monotonicity, wide codes, or missing codes at the subranging points as shown in Figure 10A, B, and C, respectively. This type of ADC should be trimmed so that drift due to aging or temperature produces wide codes at the sensitive points rather than non-monotonic or missing codes.

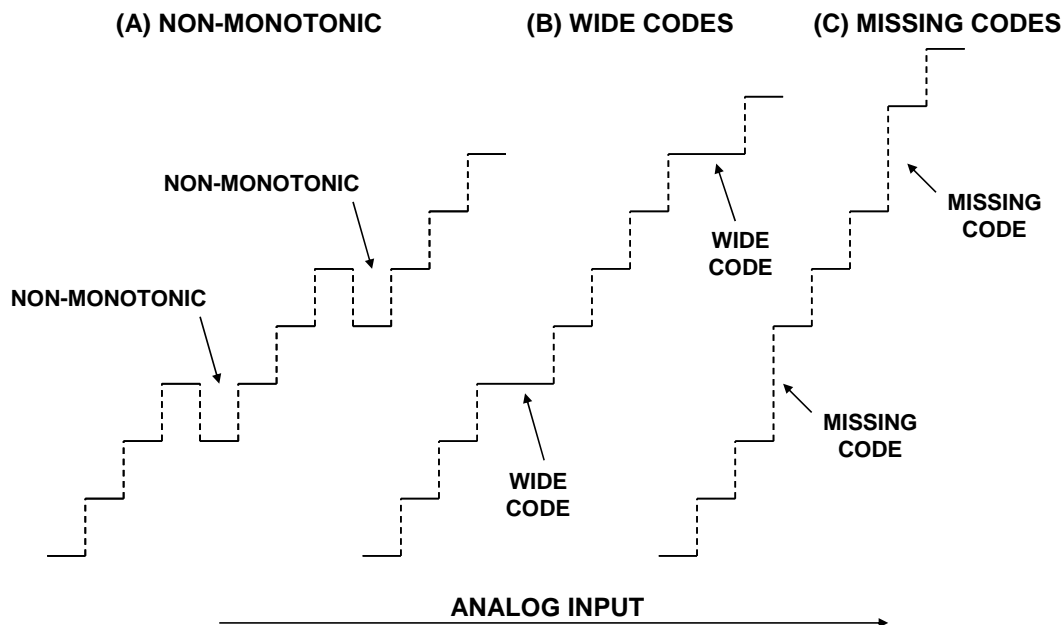


Figure 10: Errors Associated with Improperly Trimmed Subranging ADC

Defining missing codes is more difficult than defining non-monotonicity. All ADCs suffer from some inherent transition noise as shown in Figure 11 (think of it as the flicker between adjacent values of the last digit of a DVM). As resolutions and bandwidths become higher, the range of input over which transition noise occurs may approach, or even exceed, 1 LSB. High resolution wideband ADCs generally have internal noise sources which can be reflected to the input as effective input noise summed with the signal. The effect of this noise, especially if combined

with a negative DNL error, 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.

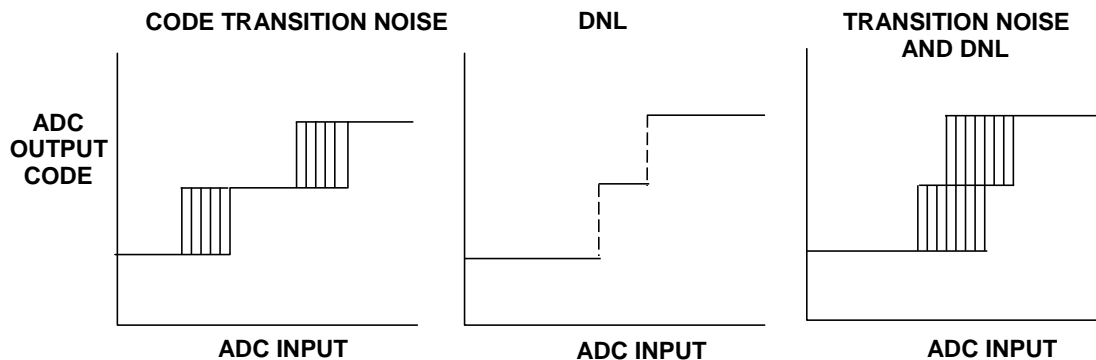


Figure 11: Combined Effects of Code Transition Noise and DNL

For low 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. A complete discussion of effective input noise is given in References 2 and 3.

REFERENCES:

1. Dan Sheingold, *Analog-Digital Conversion Handbook, 3rd Edition*, Analog Devices and Prentice-Hall, 1986, ISBN-0-13-032848-0. (the defining and classic book on data conversion).
2. Walt Kester, *Analog-Digital Conversion*, Analog Devices, 2004, ISBN 0-916550-27-3, Chapter 2 and 5. Also available as *The Data Conversion Handbook*, Elsevier/Newnes, 2005, ISBN 0-7506-7841-0, Chapters 2 and 5.
3. Walt Kester, "The Good, the Bad, and the Ugly Aspects of ADC Input Noise: Is No Noise Good Noise?" *Tutorial MT-004*, Analog Devices.

Copyright 2009, Analog Devices, Inc. All rights reserved. Analog Devices assumes no responsibility for customer product design or the use or application of customers' products or for any infringements of patents or rights of others which may result from Analog Devices assistance. All trademarks and logos are property of their respective holders. Information furnished by Analog Devices applications and development tools engineers is believed to be accurate and reliable, however no responsibility is assumed by Analog Devices regarding technical accuracy and topicality of the content provided in Analog Devices Tutorials.