The Good, the Bad, and the Ugly Aspects of ADC Input Noise—Is No Noise Good Noise?

by Walt Kester

INTRODUCTION

All analog-to-digital converters (ADCs) have a certain amount of "input-referred noise"—modeled as a noise source connected in series with the input of a noise-free ADC. Input-referred noise is not to be confused with quantization noise which only occurs when an ADC is processing an ac signal. In most cases, less input noise is better, however there are some instances where input noise can actually be helpful in achieving higher resolution. This probably doesn't make sense right now, so you will just have to read further into this tutorial to find out how SOME noise can be GOOD noise.

INPUT-REFERRED NOISE (CODE TRANSITION NOISE)

Practical ADCs deviate from ideal ADCs in many ways. Input-referred noise is certainly a departure from the ideal, and its effect on the overall ADC transfer function is shown in Figure 1. As the analog input voltage is increased, the "ideal" ADC (shown in Figure 1A) maintains a constant output code until the transition region is reached, at which point the output code instantly jumps to the next value and remains there until the next transition region is reached. A theoretically perfect ADC has zero "code transition" noise, and a transition region width equal to zero. A practical ADC has certain amount of code transition noise, and therefore a transition region width that depends on the amount of input-referred noise present (shown in Figure 1B). Figure 1B shows a situation where the width of the code transition noise is approximately one least significant bit (LSB) peak-to-peak.

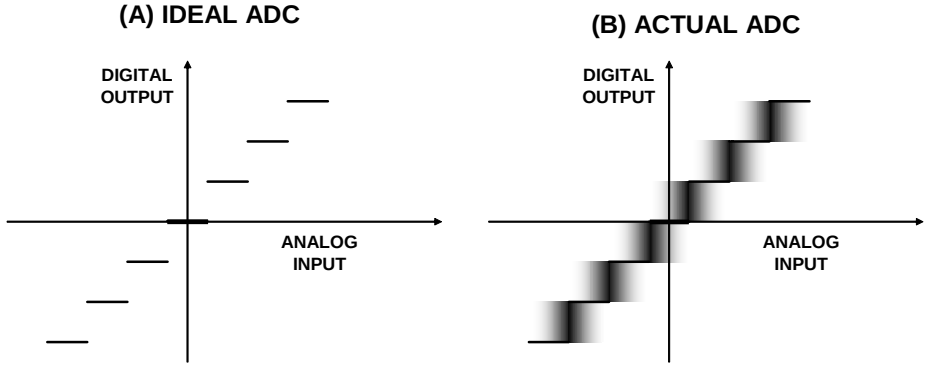

Figure 1: Code Transition Noise (Input-referred Noise) and its Effect on ADC Transfer Function

All ADC internal circuits produce a certain amount of rms noise due to resistor noise and "kT/C" noise. This noise is present even for dc-input signals, and accounts for the code transition noise. Today, *code transition noise* is most often referred to as *input-referred noise* rather than *code transition noise*. The input-referred noise is most often characterized by examining the histogram of a number of output samples when the input to the ADC is a dc value. The output of most high speed or high resolution ADCs is a distribution of codes, centered around the nominal value of the dc input (see Figure 2). To measure its value, the input of the ADC is either grounded or connected to a heavily decoupled voltage source, and a large number of output samples are collected and plotted as a histogram (sometimes referred to as a *grounded-input* histogram). Since the noise is approximately Gaussian, the standard deviation of the histogram, σ, can be calculated, corresponding to the effective input rms noise. Reference 1 provides a detailed description of how to calculate the value of σ from the histogram data. It is common practice to express this rms noise in terms of LSBs rms, although it can be expressed as an rms voltage referenced to the ADC full-scale input range.

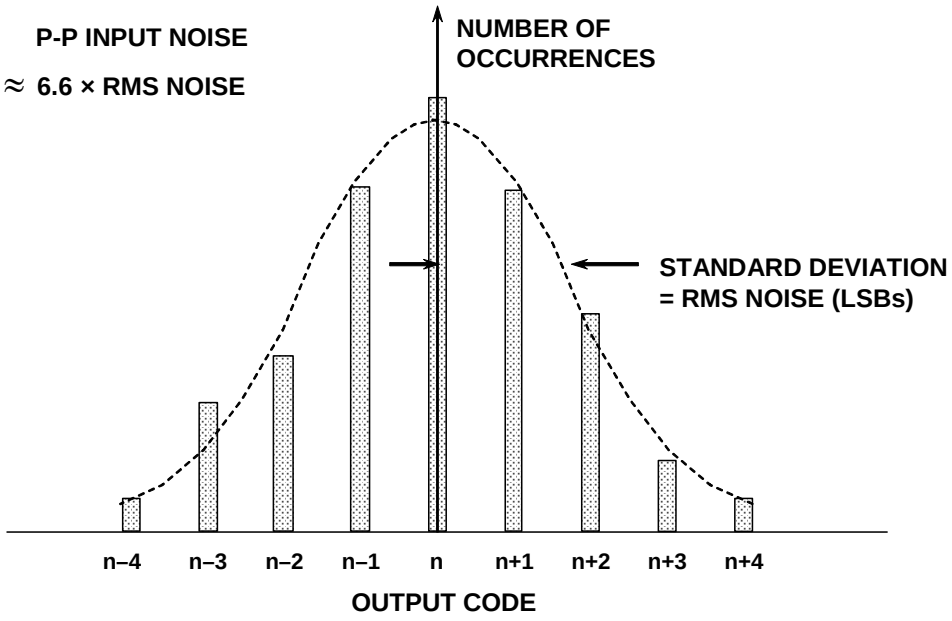

Figure 2: Effect of Input-Referred Noise on ADC "Grounded Input" Histogram for ADC with Small Amount of DNL

Although the inherent differential nonlinearity (DNL) of the ADC may cause some minor deviations from an ideal Gaussian distribution (some DNL is shown in Figure 2, for instance), it should be at least approximately Gaussian. If there is significant DNL, the value of σ should be calculated for several different dc input voltages, and the results averaged. If the code distribution has large and distinct peaks and valleys, for instance—this indicates either a poorly designed ADC or, more likely, a bad PC board layout, poor grounding techniques, or improper power supply decoupling (see Figure 3). Another indication of trouble is when the width of the distribution changes drastically as the dc input is swept over the ADC input voltage range.

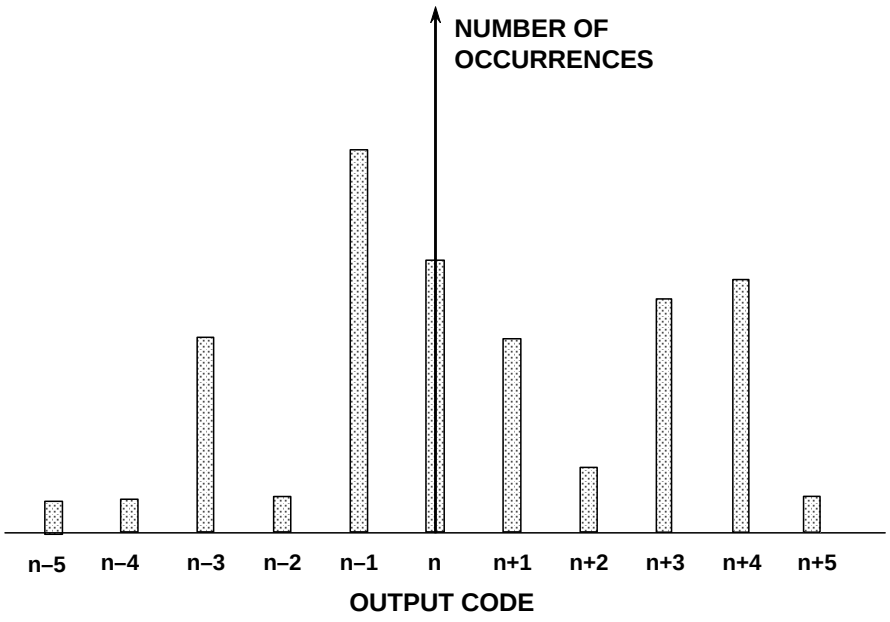

Figure 3: Grounded Input Histogram for Poorly Designed ADC and/or Poor Layout, Grounding, or Decoupling

NOISE-FREE (FLICKER-FREE) CODE RESOLUTION

The *noise-free code resolution* of an ADC is the number of bits beyond which it is impossible to distinctly resolve individual codes. The cause is the effective input noise (or input-referred noise) associated with all ADCs and described above. This noise can be expressed as an rms quantity, usually having the units of *LSBs rms*. Multiplying by a factor of 6.6 converts the rms noise into peak-to-peak noise (expressed in *LSBs peak-to-peak*). The total range (or span) of an N-bit ADC is 2^N LSBs. The total number of *noise-free counts* is therefore equal to:

$$\text{Noise - Free Counts} = \frac{2^N}{\text{Peak - Peak Input Noise (LSBs)}}. \qquad \text{Eq. 1}$$

The number of noise-free counts can be converted into noise-free code resolution by taking the base-2 logarithm as follows:

$$\text{Noise Free Code Resolution} = \log_2\left(\frac{2^N}{\text{Peak - Peak Input Noise (LSBs)}}\right). \qquad \text{Eq. 2}$$

The noise-free code resolution specification is generally associated with high-resolution sigma-delta measurement ADCs. It is most often a function sampling rate, digital filter bandwidth, and programmable gain amplifier (PGA) gain. Figure 4 shows a typical table taken from the AD7730 sigma-delta measurement ADC.

Table II. Peak-to-Peak Resolution vs. Input Range and Update Rate (CHP = 1)

			Peak-to-Peak Resolution in Counts (Bits)					
Output Data Rate	–3 dB Frequency	SF Word	Settling Time Normal Mode	Settling Time Fast Mode	Input Range = ±80 mV	Input Range = ±40 mV	Input Range = ±20 mV	Input Range = ±10 mV
50 Hz	1.97 Hz	2048	460 ms	60 ms	230k (18)	175k (17.5)	120k (17)	80k (16.5)
100 Hz	3.95 Hz	1024	230 ms	30 ms	170k (17.5)	125k (17)	90k (16.5)	55k (16)
150 Hz	5.92 Hz	683	153 ms	20 ms	130k (17)	100k (16.5)	70k (16)	45k (15.5)
200 Hz*	7.9 Hz	512	115 ms	15 ms	120k (17)	90k (16.5)	65k (16)	40k (15.5)
400 Hz	15.8 Hz	256	57.5 ms	7.5 ms	80k (16.5)	55k (16)	40k (15.5)	30k (15)

*Power-On Default

Figure 4: Noise-Free Code Resolution for the AD7730 Sigma-Delta ADC

Note that for an output data rate of 50 Hz and an input range of ±10 mV, the noise-free code resolution is 16.5 bits (80,000 noise-free counts). The settling time under these conditions is 460 ms, making this ADC an ideal candidate for a precision weigh scale application. Data such as this is available on most data sheets for high resolution sigma-delta ADCs suitable for precision measurement applications.

The ratio of the FS range to the *rms* input noise (rather than peak-to-peak noise) is sometimes used to calculate resolution. In this case, the term *effective resolution* is used. Note that under identical conditions, *effective resolution* is larger than *noise-free code resolution* by $\log_2(6.6)$, or approximately 2.7 bits.

$$\text{Effective Resolution} = \log_2\left(\frac{2^N}{\text{RMS Input Noise (LSBs)}}\right) \qquad \text{Eq. 3}$$

$$\text{Effective Resolution} = \text{Noise Free Code Resolution} + 2.7 \text{ Bits}. \qquad \text{Eq. 4}$$

Some manufacturers prefer to specify effective resolution rather than noise-free code resolution because it results in a higher number of bits—the user should check the data sheet closely to make sure which is actually specified.

INCREASING ADC "RESOLUTION" AND REDUCING NOISE BY DIGITAL AVERAGING

The effects of input-referred noise can be reduced by digital averaging. Consider a 16-bit ADC which has 15 noise-free bits at a sampling rate of 100 kSPS. Averaging two samples per output sample reduces the effective sampling rate to 50 kSPS and increases the SNR by 3 dB, and the noise-free bits to 15.5. Averaging four samples per output sample reduces the sampling rate to 25 kSPS, increases the SNR by 6 dB, and increases the noise-free bits to 16.

In fact, if we average sixteen samples per output, the output sampling rate is reduced to 6.25 kSPS, the SNR increases by another 6 dB, and the noise-free bits increase to 17. The arithmetic in the averaging must be carried out to the larger number of significant bits in order to take advantage of the extra "resolution."

The averaging process also helps "smooth" out the DNL errors in the ADC transfer function. This can be illustrated for the simple case where the ADC has a missing code at quantization level "k". Even though code "k" is missing because of the large DNL error, the average of the two adjacent codes, k – 1 and k + 1 is equal to k.

This technique can therefore be used effectively to increase the dynamic range of the ADC, at the expense of overall output sampling rate and extra digital hardware. It should be noted, however, that averaging will not correct the inherent integral nonlinearity of the ADC.

Now, consider the case of an ADC that has extremely low input-referred noise, and the histogram shows a solid code all the time. What will digital averaging do for this ADC? This answer is simple—it will do nothing! No matter how many samples we average, we will still get the same answer. However, as soon as we add enough noise to the input signal so that there is more than one code in the histogram, the averaging method starts working again. Some small amount of noise is therefore good (at least with respect to the averaging method), but the more noise present at the input, the more averaging is required to achieve the same resolution.

DON'T CONFUSE EFFECTIVE NUMBER OF BITS (ENOB) WITH "EFFECTIVE RESOLUTION" OR "NOISE-FREE CODE RESOLUTION"

Because of the similarity of the terms, *effective number of bits* and *effective resolution* are often assumed to be equal. This is not the case.

Effective number of bits (ENOB) is derived from an FFT analysis of the ADC output when the ADC is stimulated with a full-scale sinewave input signal. The rss value of all noise and distortion terms is computed, and the ratio of the signal to the noise and distortion is defined as SINAD, or S/(N+D). The theoretical SNR of a perfect N-bit ADC is given by:

$$SNR = 6.02N + 1.76dB. \qquad \text{Eq. 5}$$

The calculated value of SINAD for the ADC is substituted for SNR in Eq. 5, and the equation solved for N, yielding the ENOB:

$$\text{ENOB} = \frac{\text{SINAD} - 1.76\text{dB}}{6.02}. \qquad \text{Eq. 6}$$

The noise and distortion used to calculate SINAD and ENOB include not only the input-referred noise but also the quantization noise and the distortion terms. SINAD and ENOB are used to measure the dynamic performance of an ADC, while effective resolution and noise-free code resolution are used to measure the noise of the ADC under dc input conditions where there is no quantization noise.

USING NOISE DITHER TO INCREASE ADC SPURIOUS FREE DYNAMIC RANGE

There are two fundamental limitations to maximizing SFDR in a high speed ADC. The first is the distortion produced by the front-end amplifier and the sample-and-hold circuit. The second is that produced by non-linearity in the actual transfer function of the encoder portion of the ADC. The key to high SFDR is to minimize the non-linearity of each.

There is nothing that can be done externally to the ADC to significantly reduce the inherent distortion caused by the ADC front end. However, the differential nonlinearity in the ADC encoder transfer function can be reduced by the proper use of dither (external noise which is summed with the analog input signal to the ADC).

Dithering improves ADC SFDR under certain conditions (References 2-5). For example, even in a perfect ADC, there is some correlation between the quantization noise and the input signal. This can reduce the SFDR of the ADC, especially if the input signal is an exact sub-multiple of the sampling frequency. Summing broadband noise (about ½ LSB rms in amplitude) with the input signal tends to randomize the quantization noise and minimize this effect (see Figure 5A). In most systems, however, there is enough noise riding on top of the signal so that adding additional dither noise is not required. The input-referred noise of the ADC may also be enough to produce the same effect. Increasing the wideband rms noise level beyond approximately one LSB will proportionally reduce the ADC SNR and results in no additional improvement.

Other schemes have been developed which use larger amounts of dither noise to randomize the transfer function of the ADC. Figure 5B also shows a dither noise source comprised of a pseudo-random number generator which drives a DAC. This signal is subtracted from the ADC input signal and then digitally added to the ADC output, thereby causing no significant degradation in SNR. An inherent disadvantage of this technique is that the allowable input signal swing is reduced as the amplitude of the dither signal is increased. This reduction in signal amplitude is required to prevent overdriving the ADC. It should be noted that this scheme does not significantly improve distortion created by the front-end of the ADC, only that produced by the non-linearity of the ADC encoder transfer function.

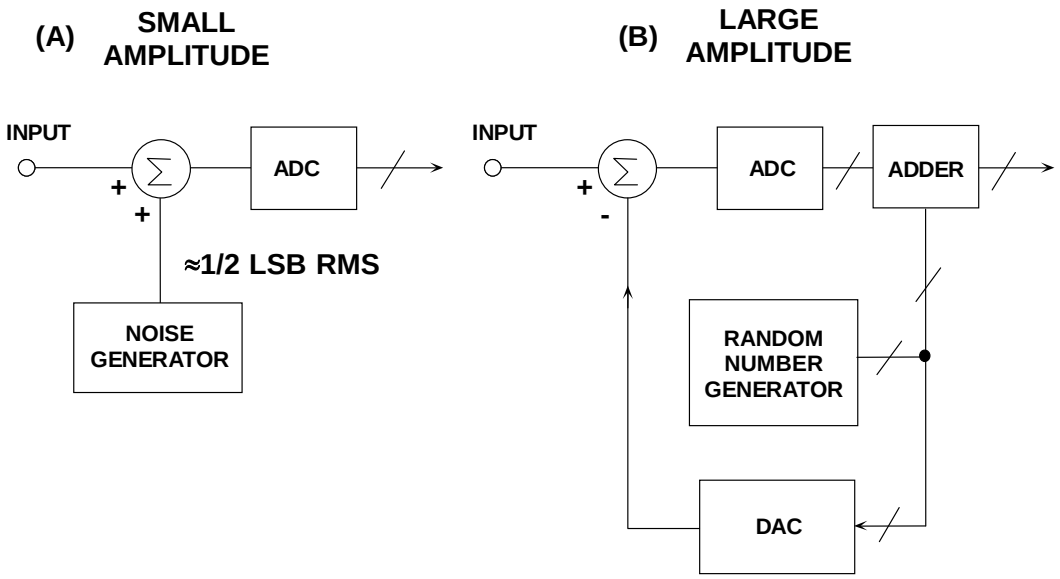

Figure 5: Using Dither to Randomize ADC Transfer Function

Another method which is easier to implement, especially in wideband receivers, is to inject a narrowband dither signal *outside the signal band of interest* as shown in Figure 6. Usually, there are no signal components located in the frequency range near dc, so this low-frequency region is often used for such a dither signal. Another possible location for the dither signal is slightly below $f_S/2$. Because the dither signal occupies only a small bandwidth relative to the signal bandwidth (usually a bandwidth of a few hundred kHz is sufficient), there is no significant degradation in SNR, as would occur if the dither was broadband.

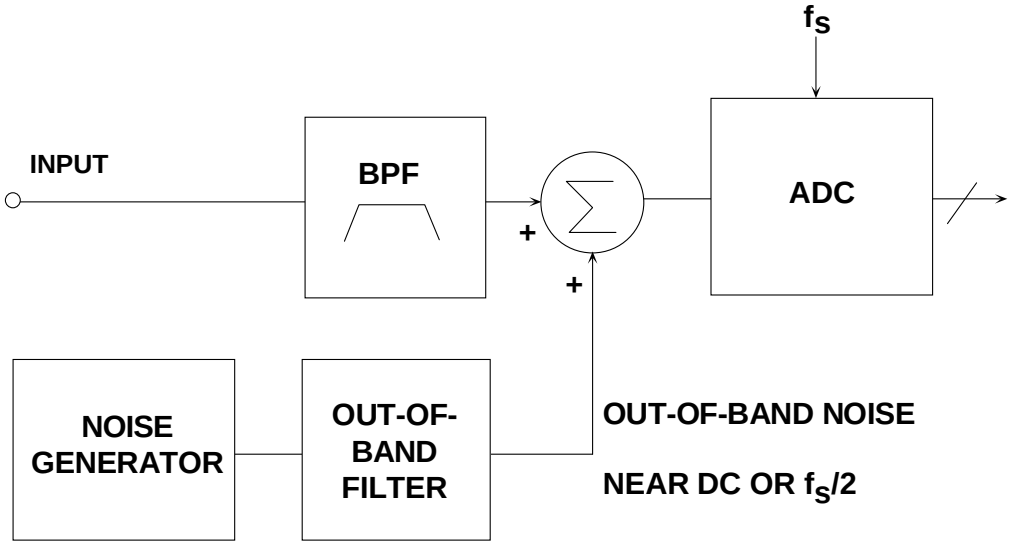

Figure 6: Injecting Out-of-Band Dither to Improve ADC SFDR

A subranging pipelined ADC, such as the AD6645 14-bit, 105 MSPS ADC, (see Figure 7), has very small differential non-linearity errors that occur at specific code transition points across the ADC range. The AD6645 uses a 5-bit ADC (ADC1) followed by a 5-bit ADC2 and a 6-bit ADC3. The only significant DNL errors occur at the ADC1 transition points—the second and third stage ADC DNL errors are minimal. There are 2^5 = 32 decision points associated with ADC1, and they occur every 68.75-mV (2^9 = 512 LSBs) for a 2.2-V fullscale input range. Figure 8 shows a greatly exaggerated representation of these nonlinearities.

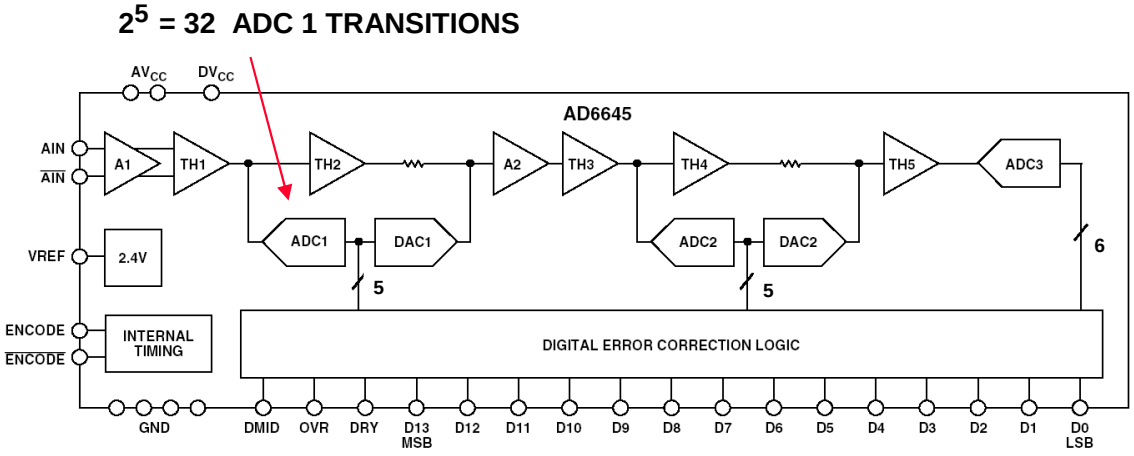

Figure 7: AD6645 14-Bit, 105 MSPS ADC Simplified Block Diagram

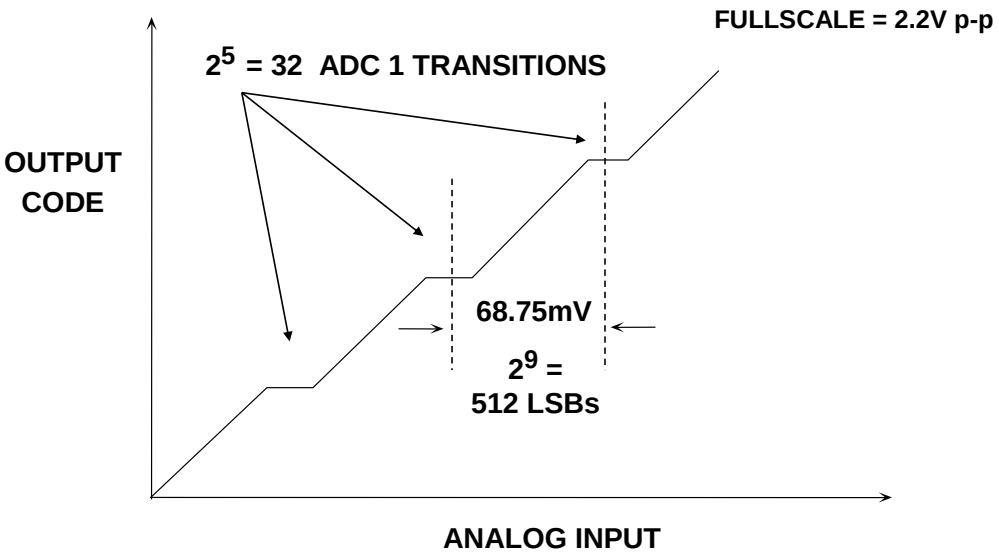

Figure 8: AD6645 Subranging Point DNL Errors (Exaggerated)

The distortion components produced by the front end of the AD6645 up to about 200-MHz analog input are negligible compared to those produced by the encoder. That is, the static non-linearity of the AD6645 transfer function is the chief limitation to SFDR.

The goal is to select the proper amount of out-of-band dither so that the effect of these small DNL errors are *randomized* across the ADC input range, thereby reducing the average DNL error. Experimentally, it was determined that making the peak-to-peak dither noise cover about two ADC1 transitions gives the best improvement in DNL. The DNL is not significantly improved with higher levels of noise. Two ADC1 transitions cover 1024 LSBs peak-to-peak, or approximately 155 LSBs rms (peak-to-peak gaussian noise is converted to rms by dividing by 6.6).

The first plot shown in Figure 9 shows the undithered DNL over a small portion of the input signal range. The horizontal axis has been expanded to show two of the subranging points which are spaced 68.75 mV (512 LSBs) apart. The second plot shows the DNL after adding 155 LSBs rms dither. This amount of dither corresponds to approximately –20.6 dBm. Note the dramatic improvement in the DNL.

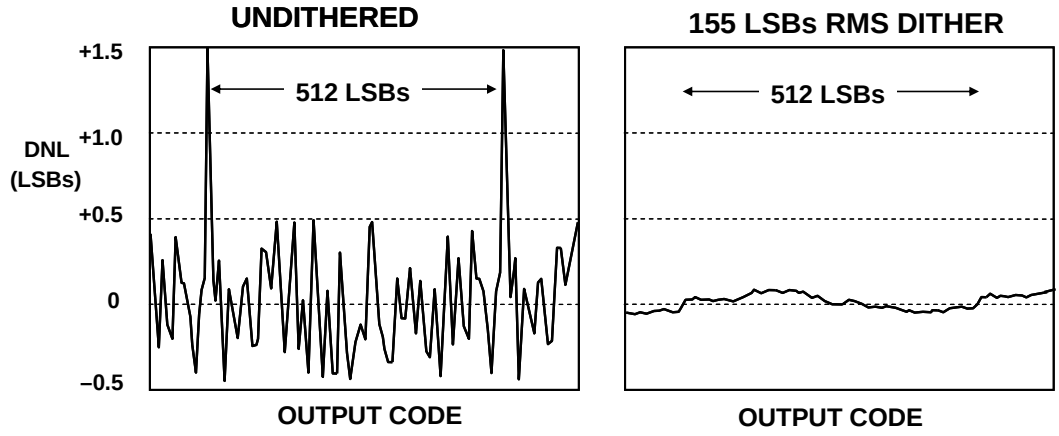

Figure 9: AD6645 Undithered and Dithered DNL

Dither noise can be generated in a number of ways. Noise diodes can be used, but simply amplifying the input voltage noise of a wideband bipolar op amp provides a more economical solution. This approach has been described in detail (References 3, 4, and 5) and will not be repeated here.

The dramatic improvement in SFDR obtained with out-of-band dither is shown in Figure 10 using a deep (1,048,576-point) FFT, where the AD6645 is sampling a –35-dBm, 30.5-MHz signal at 80 MSPS. Note that the SFDR without dither is approximately 92 dBFS compared to 108 dBFS with dither, representing a 16-dB improvement!

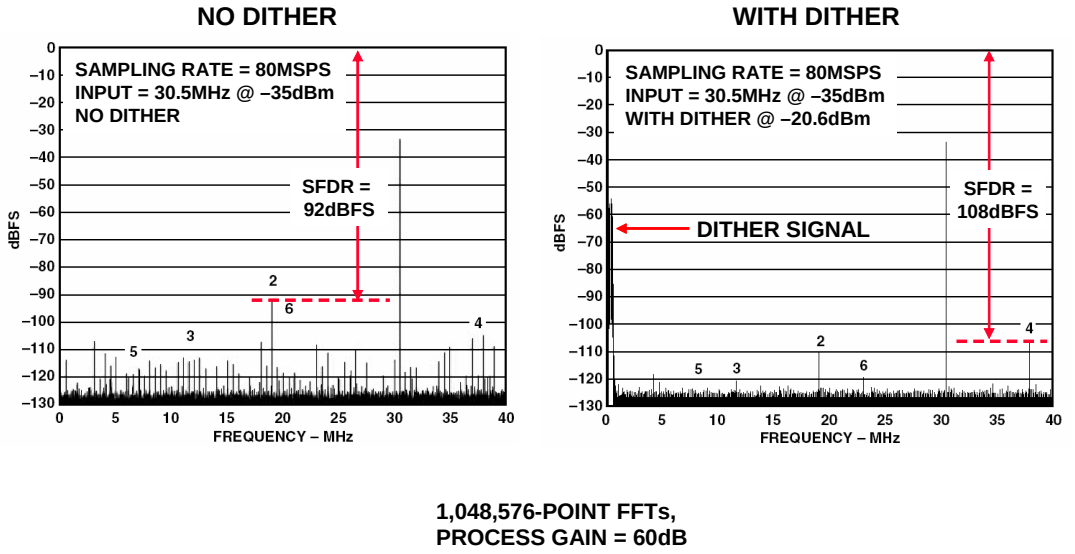

Figure 10: AD6645 Undithered and Dithered SFDR FFT Plot

The AD6645 ADC was introduced by Analog Devices in 2000. Until recently, it represented the ultimate in SFDR performance. Since its introduction, improvements in both process technology and circuit design have resulted in even higher performance ADCs such as the AD9444 (14-bits @ 80 MSPS), AD9445 (14-bits @ 105/125 MSPS), and the AD9446 (16-bits @ 80/100 MSPS). These ADCs have very high SFDR (typically greater than 90 dBc for a 70-MHz full-scale input signal) and low DNL. Still, the addition of an appropriate out-of-band dither signal can improve the SFDR under certain input signal conditions.

Figure 11 shows the AD9444 (14-bits @ 80MSPS) FFT with and without dither. Note that under these input conditions, the addition of dither improves the SFDR by 25 dB. The data was taken using the ADIsimADC program and the AD9444 model.

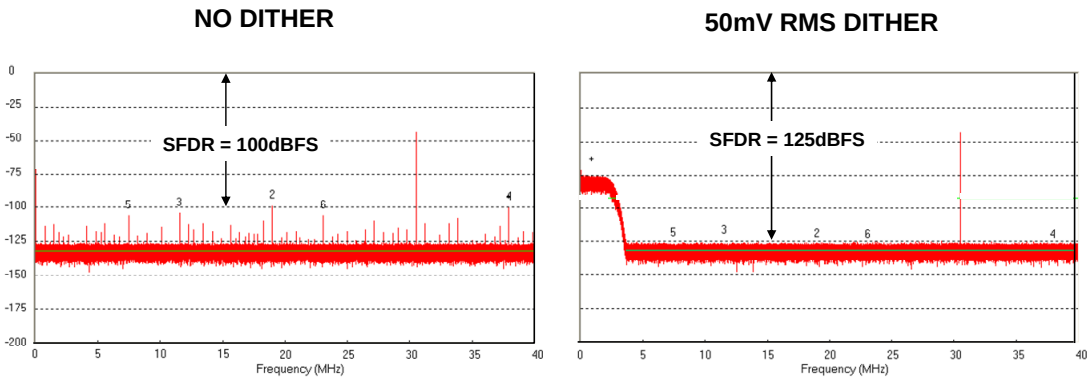

Figure 11: AD9444, 14-Bit, 80MSPS ADC, f_s = 80MSPS, f_{in} = 30.5MHz, Signal Amplitude = –40dBFS

Even though the results shown in Figures 10 and 11 are fairly dramatic, it should not be assumed that the addition of out-of-band noise dither will always improve the SFDR of the ADC or under all conditions. As mentioned earlier, dither will not improve the linearity of the front end circuits of the ADC. Even with a nearly ideal front end, the effects of dither will be highly dependent upon the amplitude of the input signal as well as the amplitude of the dither signal itself. For example, when signals are near the fullscale input range of the ADC, the integral nonlinearity of the transfer function may become the limiting factor in determining SFDR, and dither will not help. In any event, the data sheet should be studied carefully—in some cases dithered and undithered data may be shown along with suggestions for the amplitude and bandwidth. Dither may be a built-in feature of newer IF sampling ADCs.

SUMMARY

In this discussion we have shown that all ADCs have some amount of input-referred noise. In precision, low frequency measurement applications, this noise can be reduced by digitally averaging the ADC output data at the expense of lower sampling rates and additional hardware. The resolution of the ADC can actually be increased by this averaging process, however integral nonlinearity errors are not improved. A small amount of input-referred noise is needed to increase the resolution by the averaging technique, however too much noise requires a large number of samples in the average, and a point of diminishing returns is reached.

In certain high speed ADC applications, the addition of the proper amount of out-of-band noise dither can improve the DNL of the ADC and increase the spurious free dynamic range (SFDR). However, the effectiveness of dither on the SFDR is highly dependent upon the characteristics of the particular ADC under consideration.

ACKNOWLEDGEMENTS

The author would like to thank Bonnie Baker of Microchip Technology and Alain Guery of Analog Devices for their thoughtful inputs to this tutorial.

REFERENCES

1. Steve Ruscak and Larry Singer, "Using Histogram Techniques to Measure A/D Converter Noise," *Analog Dialogue*, Vol. 29-2, 1995.

2. Brad Brannon, "Overcoming Converter Nonlinearities with Dither," *Application Note AN-410*, Analog Devices, 1995.

3. Walt Jung, "Wideband Noise Generator," Ideas for Design, *Electronic Design*, October 1, 1996.

4. Walter G. Jung, *Op Amp Applications*, Analog Devices, 2002, ISBN 0-916550-26-5, Also available as *Op Amp Applications Handbook*, Elsevier/Newnes, 2005, ISBN 0-7506-7844-5..

5. Walt Kester, "Add Noise Dither to Blow Out ADCs' Dynamic Range," *Electronic Design, Analog Applications Supplement*, November 22, 1999, pp. 20-26.

6. Bonnie Baker, "Sometimes, Noise Can Be Good," *EDN*, February 17, 2005, p. 26.

7. Hank Zumbahlen, *Basic Linear Design*, Analog Devices, 2006, ISBN: 0-915550-28-1. Also available as *Linear Circuit Design Handbook*, Elsevier-Newnes, 2008, ISBN-10: 0750687037, ISBN-13: 978-0750687034. Chapter 12

8. Walt Kester, *Analog-Digital Conversion*, Analog Devices, 2004, ISBN 0-916550-27-3, Chapter 6. Also available as *The Data Conversion Handbook*, Elsevier/Newnes, 2005, ISBN 0-7506-7841-0, Chapter 6