DIGITAL TECHNIQUES FOR IMPROVING THE ACCURACY OF DATA CONVERTERS

UN-KU MOON AND GABOR C. TEMES, OREGON STATE UNIVERSITY
JESPER STEENSGAARD, COLUMBIA UNIVERSITY

ABSTRACT

This article provides a tutorial overview of some recently developed methods for enhancing the accuracy and linearity of data converters (analog-to-digital as well as digital-to-analog) by introducing auxiliary digital circuitry which calibrates, cancels and/or corrects the errors introduced by the unavoidable inaccuracy of the analog components used in the conversion. Simple but practical examples are used to illustrate the various improvement techniques. Using the described methods, the inherent accuracy and linearity of the data conversion can be extended from about 10 bits to 16 bits and more.

The operation of communication and entertainment systems is increasingly based on digital signal processing (DSP), while the physical signals needed to be handled at the input and output nodes of these systems remain continuous-time analog ones. Hence, such a system typically needs an analog-to-digital converter (ADC) at its input end, and a digital-to-analog converter (DAC) at its output end. These converters contain both digital and analog circuitry, and their speed and accuracy is usually limited by that of the analog components. As the speed and accuracy of the internal DSP increases, the corresponding requirements on these converters also become harder to meet. Also, while state-of-the-art integrated circuit (IC) processes help with the design of faster analog as well as digital circuits, they are detrimental rather than helpful in meeting the analog accuracy specifications due to their reduced dimensions and supply voltages.

In most data converters, the accuracy is limited by the actual matching accuracy of nominally equal-sized analog circuit elements (resistors, or capacitors, or current sources). Thus, to perform an 18-bit digital-to-analog conversion as needed in, say, digital audio applications, the required matching error of the components used cannot exceed 2^{-18}, or about four parts per million. Such accuracy is difficult if not impossible to obtain, even if an expensive and time-consuming process such as on-chip trimming is used. Hence, alternative techniques have been developed, which embed DSP blocks in the analog circuit to help reduce the effects of the limited analog matching accuracy on the overall performance of the data converters. The best known and most widely used example of this process is the delta-sigma ($\Delta\Sigma$) data converter, which has been discussed in detail in recent literature [1]. These converters contain an internal quantizer of much lower resolution than that of the overall conversion, and use feedback to achieve high in-band accuracy.

In this article, alternative techniques are discussed and illustrated using some very simple but practical converter circuits. They can be classified into four categories: analog correction, digital correction, error cancellation, and spectral error shaping. In analog correction, an analog quantity (voltage, current, capacitance, and so on) is adjusted under digital control to regain the required accuracy. In digital correction, a digital signal is modified so as to rectify the error made due to analog inaccuracies. In error cancellation, the conversion process is performed such that the unknown error enters twice, with opposite polarities, and is hence cancelled. Finally, in spectral error shaping (also called mismatch shaping) the conversion process modifies the spectrum of the conversion error signal (which is unknown to the designer) so as to suppress its energy in the signal band. The resulting corrected stages can be used as stand-alone data converters, or as internal blocks in pipeline or $\Delta\Sigma$ ADCs or DACs.

The correction algorithms may be performed only once, at power-up, or they can take place continuously during the conversion process. The latter is usually preferable, since it can also correct for time-varying inaccuracies caused, for example, by temperature variations.

In the following sections, these techniques will be described and illustrated with simple examples. Even though the matched elements in converters may also be resistors or current sources, we shall assume for simplicity that the operation is based on matched capacitors (i.e., that switched-capacitor (SC) circuitry is used). The algorithms and general techniques remain, however, applicable to other circuit realizations.

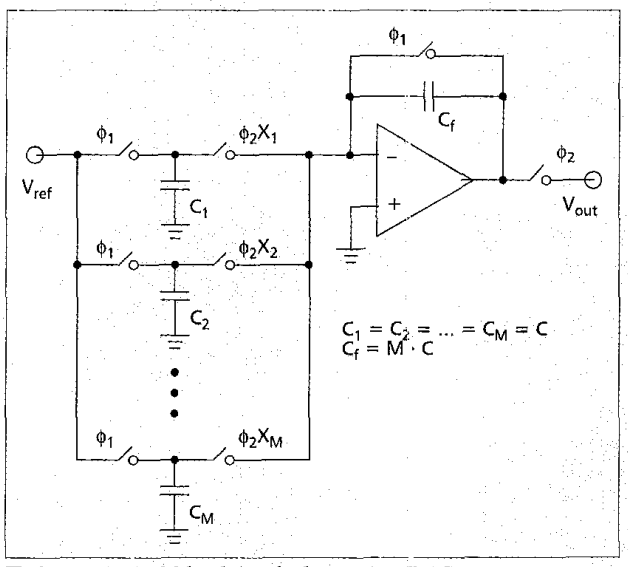
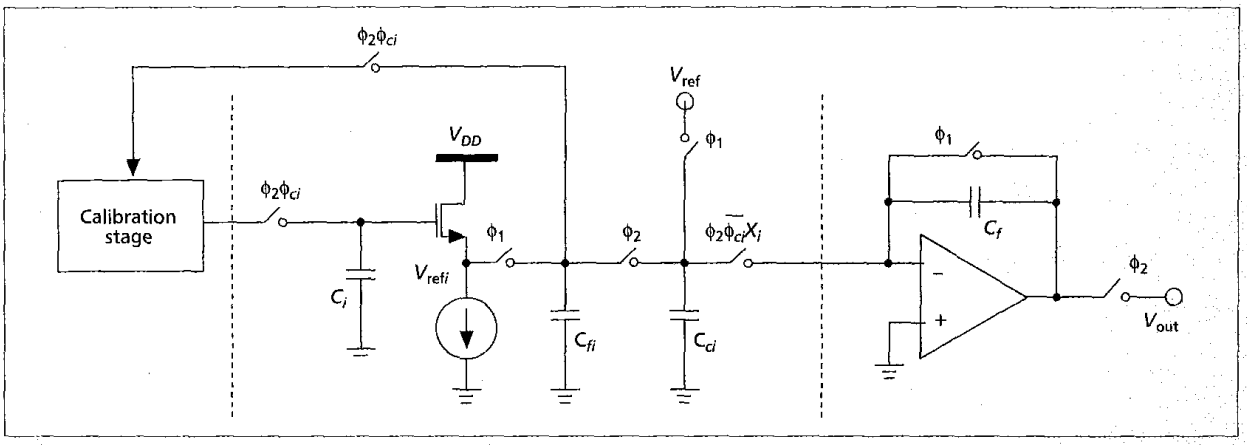


Figure 1. *An M-level switched-capacitor DAC.*

0163-6804/99/$10.00 © 1999 IEEE



■ **Figure 2.** *The DAC of Fig. 1 with analog mismatch correction.*

Analog Correction Techniques

Figure 1 shows the conceptual diagram of a DAC constructed using SC circuitry. The operation of the circuit is as follows. The input binary word is converted into a thermometer code with bits $x_1, x_2, \ldots, x_M$ such that if the integer value of the input word is m, the bits $x_1, x_2, \ldots, x_m$ are 1, and the rest are 0. During the reset phase ($\phi_1=1$), the feedback capacitor C_f is discharged and all input capacitors are charged to the reference voltage V_{ref}. Next, during the conversion phase ($\phi_2=1$), the first m input capacitors are discharged into C_f, resulting in an output voltage $V_{out} = -(mC/C_f)V_{ref} = -(m/M)V_{ref}$.

Since in practice the nominally equal-valued input capacitors will have different capacitances, a fixed nonlinearity will be introduced, which will cause harmonic distortion. For a well designed integrated circuit, a random matching error of 0.1 percent may be expected; this is likely to cause a total harmonic distortion that is about 0.1 percent of the full-scale signal. For DACs with more than 10-bit accuracy, this is unacceptable.

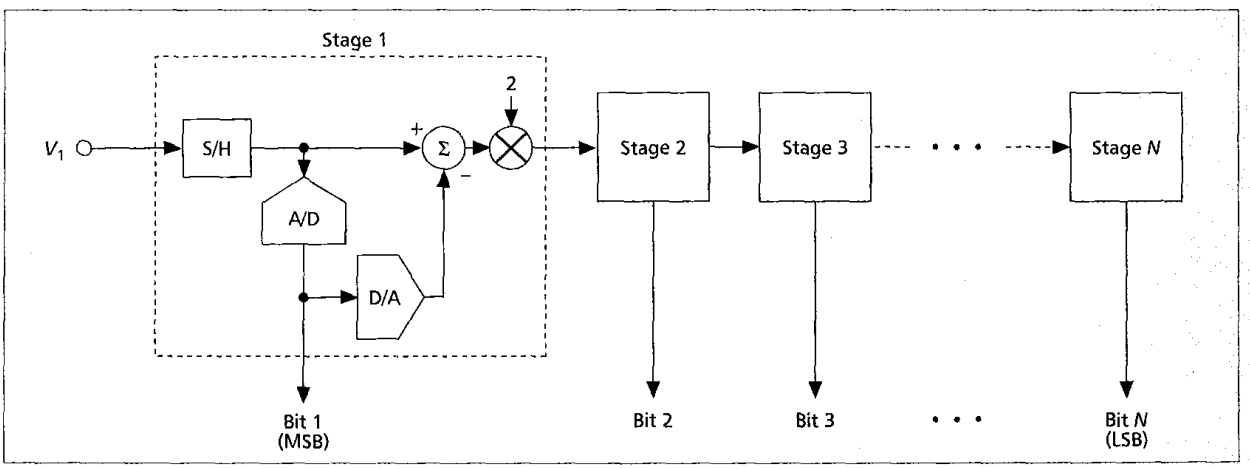
Analog correction of the matching error may be achieved by using the system shown in Fig. 2, where each capacitor C_i is split into a coarse part and a fine part, C_{ci} and C_{fi}, respectively, and a separate buffered reference voltage V_{refi} is introduced for each C_{fi} [2]. When clock phase 2 is high ($\phi_2=1$), capacitors C_{ci} and C_{fi} are discharged into C_f if $x_i=1$, otherwise they hold their charges. A calibration stage, consisting of a transconductor and a reference capacitor C_{ref}, is used to readjust the V_{refi} sequentially when the i-th calibration clock phase ϕ_{ci} is high, so as to make the combined charges stored in C_{ci} and C_{fi} equal to $C_{ref}V_{ref}$. To replace the capacitor being calibrated, an extra set of C_c and C_f is also needed. The resulting conversion accuracy can then be as high as 15 bits.

The process is similar to that proposed earlier for current-mode DACs by Schouwenaars *et al.* [3], in which 16 bidirectional current sources were used in the 4-bit internal DAC of a 20-bit DS DAC. A level of 90 dB S/(N+THD) was achieved using a bidirectional current reference which was copied into each current source sequentially.

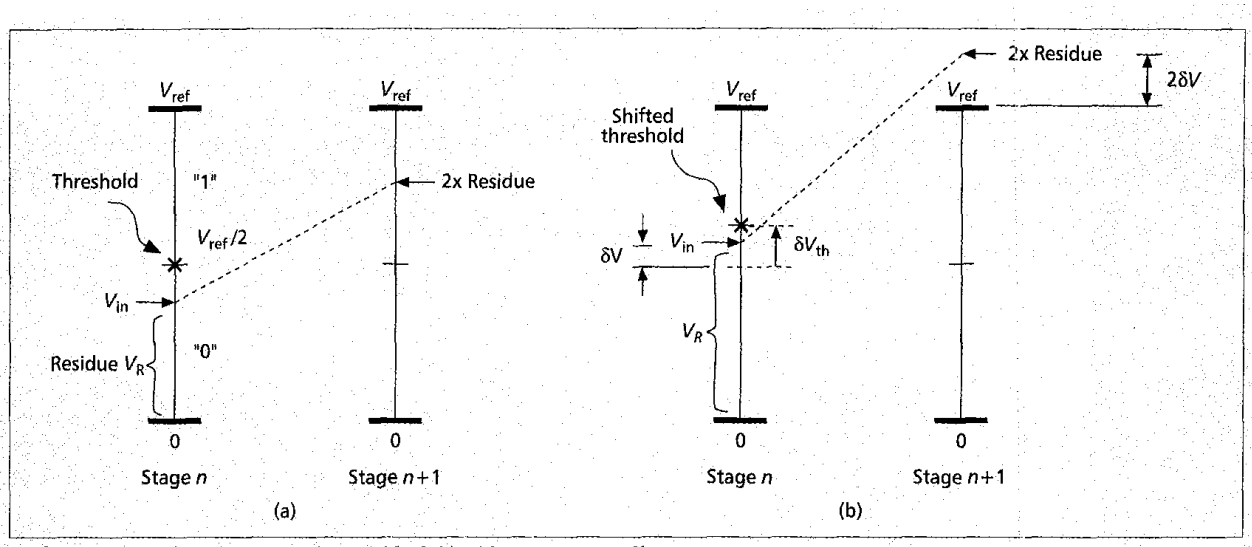
Another example of (digitally aided) analog error correction is described for a binary-weighted SC DAC embedded in an ADC by Lee *et al.* [4]. It performs a calibration cycle at power-up, stores the capacitance errors digitally, and then provides an analog correction signal delivered by a calibration DAC for any input code.

Digital Redundancy/Correction and Calibration

In this category of digital error compensation, we shall first discuss digital redundancy (often referred to as digital correction) which compensates for nonzero comparator offsets leading to code errors in data converters, and then digital calibration which measures and compensates for other analog errors such as capacitor mismatch errors and finite op-amp



■ **Figure 3.** *An N-bit pipeline ADC with 1-bit/stage resolution.*



■ **Figure 4.** *Pipeline stage operation: a) ideal; b) with comparator offset.*

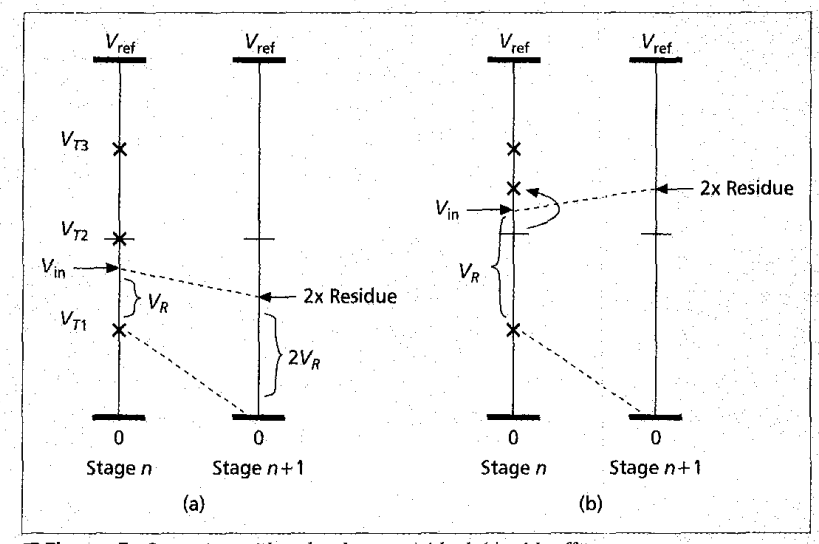
gain in the digital domain. These two digital techniques require minor architectural modifications (which may be considered analog domain modifications), but a commonality exists in that the analog errors are compensated during the postprocessing in the digital domain.

Consider a practical and simple form of an A/D converter to illustrate the digital redundancy/correction. First observe an N-bit pipeline ADC with 1-bit per stage resolution (Fig. 3). Each stage provides one bit of the output, starting with the MSB derived by stage 1, and passes on an analog residue voltage to the next stage for processing. Included in each of the 1-bit stages is a comparator (1-bit A/D), with a reference voltage $V_{ref}/2$ for a unipolar input range from 0 to V_{ref}. The operation is shown in Fig. 4a. For a given input voltage level, depending on whether the input is above or below the $V_{ref}/2$ reference, the bit "1" or "0" results. Then the residue voltage ($V_{in} - V_{ref}/2$ or $V_{in} - 0$) is entered into the following stage with 2x amplification. If the input sample is below $V_{ref}/2$, the amplified residue voltage transferred to the following stage is $2V_{in}$; if the input sample is above $V_{ref}/2$, the doubled residue voltage transferred to the following stage is $2\,(V_{in} - V_{ref}/2) = 2V_{in} - V_{ref}$. For a bipolar converter with a $-V_{ref}$ to $+V_{ref}$ input range and a comparator threshold at 0 volt, the 2x residue voltage is $2V_{in} + V_{ref}$ and $2V_{in} - V_{ref}$ for input signals below and above the comparator threshold, respectively.

Assuming again a 0 to V_{ref} (unipolar) input range, a problem arises when the comparator level is not exactly at $V_{ref}/2$. If we assume that the comparator threshold is above $+V_{ref}/2$ by an error δV_{th}, an input signal that is above $V_{ref}/2$ (by δV) but below the comparator threshold will still be resolved as a "0." The 2x amplified residue voltage will then be $2V_{in} = 2(V_{ref}/2 + \delta V) = V_{ref} + 2\delta V$. This condition is illustrated in Fig. 4b. As shown in the figure, the 2x residue voltage is out of the input range of the next stage. The input of that stage will then be saturated, and this leads directly to a critical code error simply due to the significant comparator threshold or reference offset.

Figure 5 illustrates the general concept of the digital redundancy (correction) architecture modification that can overcome this problem. By creating a higher resolution for each stage (two bits in this case), but utilizing a lower resolution (one bit here, corresponding to a 2x residue amplification, instead of 4x), the saturation at the input of the following stage is avoided. The use of redundant higher resolution in each stage allows up to +/–0.5 LSB voltage for comparator offsets. This redundancy is taken into account and corrected in the digital domain.

Perhaps the most commonly used implementation of the digital redundancy is the "1.5-bit" per stage pipeline architecture [5]. This will be explained (for a bipolar $-V_{ref}$ to $+V_{ref}$ input range) in the following. The diagram of Fig. 6a shows the threshold levels V_{t1}, V_{t2}, and V_{t3} for a 2-bit bipolar ADC; Fig. 6b shows the thresholds for a 1.5-bit bipolar ADC. The 1.5-bits/stage structure has only two comparator reference voltages at $-V_{ref}/4$ and $V_{ref}/4$, resulting in three possible output codes as shown in Fig. 6b. In comparison to the 2-bit stage illustrated in Fig. 6a, the operation of the converter of Fig. 6b resembles the same 2-bit stage with a fixed input signal offset of $-V_{ref}/4$. In other words, all comparator thresholds are shifted by $+V_{ref}/4$. The comparator with the $V_{ref}/2$ threshold is eliminated as it is unnecessary for bipolar operation.



■ **Figure 5.** *Operation with redundancy: a) ideal; b) with offset.*

The net improvement is the requirement of only two comparators per stage instead of three, without losing the important benefit of digital redundancy.

Consider now the example shown in Fig. 7. In this 1.5-bits per stage structure, the three-level operation results in three possible residue conditions:

- For an input less the $-V_{ref}/4$ threshold, the 2x residue voltage is $2V_{in}+V_{ref}$.
- For an input between the $V_{ref}/4$ and $-V_{ref}/4$ thresholds, the 2x residue voltage is $2V_{in}$.
- For an input greater than $+V_{ref}/4$, the 2x residue voltage is $2V_{in} - V_{ref}$.

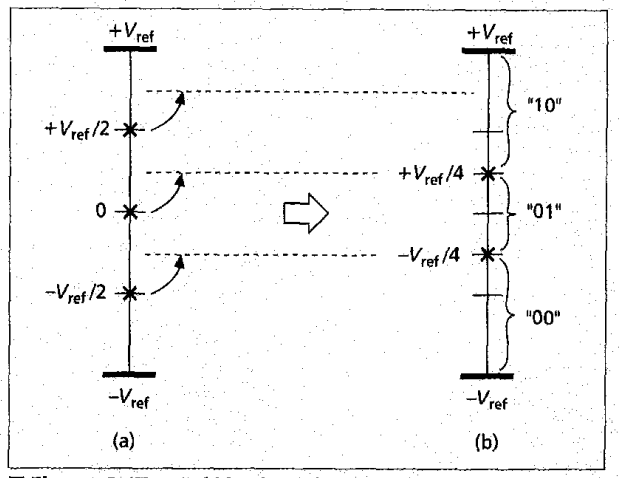
As shown in the figure, the 2x residue generated in the $\pm V_{ref}/4$ region maps into $\pm V_{ref}/2$ in the following stage; the $-V_{ref}$ to $-V_{ref}/4$ region maps into $-V_{ref}$ to $V_{ref}/2$; and the $+V_{ref}/4$ to $+V_{ref}$ region maps into $-V_{ref}/2$ to $+V_{ref}$. The signal/residue propagations shown by the shaded regions illustrate how residues can take very different paths and still result in the same digital code after digital correction is applied.

The above discussion was aimed at providing an understanding of the general concept of digital correction. A more general and rigorous description of digital redundancy can be found in [6, 7].

To introduce the concept of digital calibration, we next consider the accuracy of the 2x residue multiplication. In reality, analog circuit imperfections such as capacitor mismatches, finite opamp gain, opamp dc offset, etc., affect the accuracy of the 2x multiplication as well as the accuracy of $\pm V_{ref}$. In addition, a fixed error term may be injected into the signal during the 2x residue multiplication process.

For the 1.5-bit/stage architecture already discussed, an SC circuit often used to perform residue amplification is shown in Fig. 8a. Ideally, $C_1 = C_2$. As shown in Fig. 8b, during phase 1, the input signal is sampled onto equal-sized capacitors C_1 and C_2. During phase 2, as shown in Fig. 8c, capacitor C_1 is directly connected to the output, while capacitor C_2 is connected to $+V_{ref}$, $-V_{ref}$, or 0, depending on the output bits. For the $+V_{ref}$ connection, the charge transfer from C_2 to C_1 results in the output voltage $V_{in} + (V_{in} - V_{ref})(C_2/C_1) = 2V_{in} - V_{ref}$. The $-V_{ref}$ and 0 connections result in outputs of $2V_{in} + V_{ref}$ and $2V_{in}$, respectively. Thus, under ideal conditions, this simple SC circuit generates $2V_{in}$ + (V_{ref}, $-V_{ref}$, or 0), as required. Because of analog circuit imperfections, however, the actual 2x residues produced will be inaccurate. A calibration process that allows a correction is illustrated in Fig. 9, which shows the equivalent circuit of three stages in the pipeline, with parameters α_1, α_2, α_3 representing the coefficient error due to capacitor mismatch, the added inputs o_1, o_2, o_3 representing op-amp offset, and δ_1, δ_2, δ_3 representing the effect of finite op-amp gain. The ADC at the output represents the following stages, which at this point are assumed to realize an ideal ADC converter.

To perform the calibration, a known voltage (e.g., V_{ref}) is injected at the terminal marked V_{cal-1}, and the voltages at nodes V_i, V_{cal-2}, and V_{cal-3} are all set to 0 V. The resulting output is then converted into digital form by the "ideal" converter. Repeating this process with $V_{cal-2} = V_{ref}$ and all other voltages zero, then with $V_{cal-3} = V_{ref}$ and all others zero, etc., provides a set of data from which the conversion errors may be calculated and corrected. In a typical implementation [8], the calibration is usually performed one stage at a time (V_{cal-1} first), and the measured error is taken into account when the next stage (V_{cal-2} in this example) is calibrated. Thus, each new calibration takes into account the previously measured error in the lower bits.



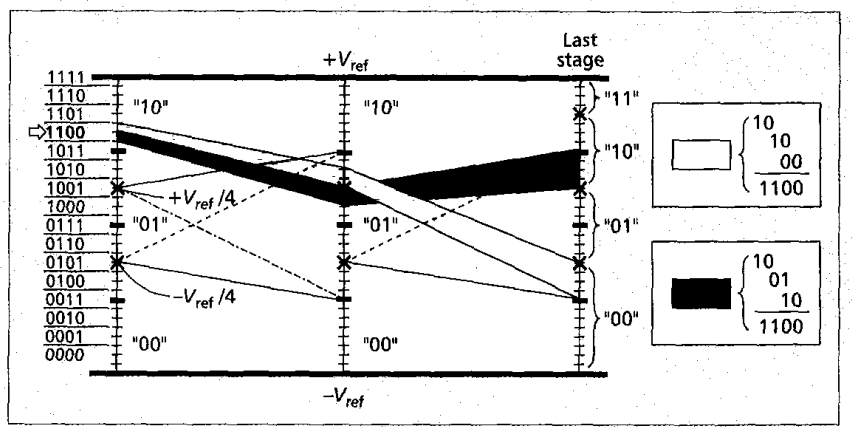
■ **Figure 6.** *Threshold levels; a) for a 2-bit bipolar ADC stage; b) for 1.5-bit bipolar stage.*

Obviously, the "ideal" ADC formed by the LSB stages is in fact imperfect; typically, it may provide 8- or 9-bit accuracy. Even so, both simulations and measured results from IC implementations have shown that the final calibrated converter is able to achieve an accuracy which is about two bits lower than the total number of bits generated. For example, for a 16-bit pipeline ADC, when the "ideal" ADC is formed from the last 8-bit stages, and the eight 1.5-bit MSB stages are calibrated in this manner, an overall resolution of about 16 – 2 = 14 bits can be achieved.

Error Cancellation Techniques

Error cancellation techniques are similar to analog calibration in the sense that analog quantities (charge, voltage, etc.) are manipulated to achieve the cancellation. In other words, although the switching operations are digitally controlled, the manipulation of the errors and their cancellation occur in the analog domain.

Perhaps the most widely used form of error cancellation is the correlated double sampling (CDS) technique [9], which



■ **Figure 7.** *Residue ranges for a 1.5-bit ADC stage.*

can cancel op-amp offset voltages and enhance the effective gain of op-amps. It is well described in the literature. However, it is also possible to cancel the effects of capacitor mismatch error, on which we focus our attention in this article.

Consider the DAC stage containing two equal-valued capacitors, shown in Fig. 10 [10], where $x(n,k)$ denotes the k-th bit of the n-th input word. Its operation under ideal conditions is as follows. The digital input words are entered serially, with the least significant bit (LSB) first. Before each word enters, both capacitors are discharged by the reset switches. Then, when $\phi_1=1$, C_1 is charged to a voltage V_{ref} or 0, depending on the LSB. Next, at $\phi_2=1$, C_1 and C_2 share charges. Afterward, C_1 is disconnected from C_2, and again is charged to V_{ref} of 0, depending on the value of the second LSB. This procedure is repeated for each bit, until the MSB has been processed. At this point the charge stored in both C_1 and in C_2, and hence (for linear capacitors) the voltage across them, represents the converted value of the input digital word.

In practice, the capacitors used cannot be made exactly equal, and hence the conversion becomes inaccurate. This introduces a deterministic nonlinearity into the process, which gives rise to harmonic distortion.

We can quantify the finite capacitor matching property by defining the error coefficient $\alpha = (C_1 - C_2)/(C_1 + C_2)$ for the nominally matched capacitors C_1 and C_2. Analyzing in detail the charge transfers that occur for the N clock cycles, it can be shown that the first-order relative error of the final DAC output (voltage on the capacitors after N clock cycles) is given by

$$Error \approx \frac{\alpha}{2} \sum_{k=1}^{N} \left[2^{k-N} x(k) - \sum_{j=1}^{k-1} 2^{j-N} x(j) \right], \qquad (1)$$

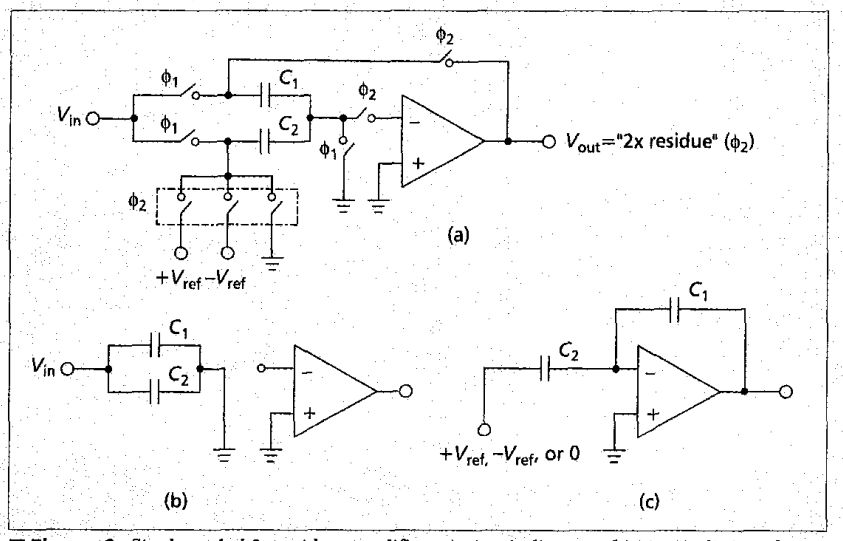
where k is the index for bits from $k = 1$ (LSB) to $k = N$ (MSB) for the N clock cycles, and the $x(k)$ are the input bits (1 or 0).

A simple way to perform capacitor mismatch error cancellation is to repeat the conversion for the same input word with the roles of capacitors C_1 and C_2 interchanged. This changes the sign of α, while leaving the rest of the formula giving *Error* unchanged. Hence, when the two outputs obtained in the two conversions for the same input word are added together, the first-order effects of the capacitor mismatch error cancel. Thus, at the cost of doubling the conversion time, the accuracy is much enhanced. Capacitor mismatch error cancellation schemes using this property were suggested for a number of different architectures [11, 12].

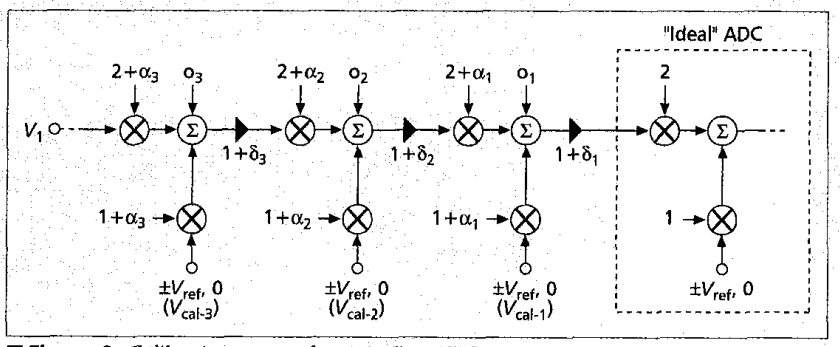
Spectral Error Shaping

In spectral error shaping, the error signal generated by the mismatch of nominally equal-valued elements gets filtered so as to suppress its in-band spectral energy. Since usually first- or second-order filtering is used, this technique will be effective only if the bandwidth f_b of the signal band occupies only a relatively small part of the 0 to $f_s/2$ range, where f_s is the sampling frequency, i.e., if the oversampling ratio $R = f_s/(2f_b)$ is much greater than 1. The filtering action can be obtained by appropriately choosing the matched elements participating in the conversion of each signal sample. To illustrate the process, consider again the circuit shown in Fig. 1 for $M = 4$. As described above, if the input data are converted into a thermometer code, then for an input having an integer value $m = 3$, normally the first three capacitors C_1, C_2, and C_3 are discharged during the conversion phase into C_f. Since these elements are mismatched, the resulting output voltage V_{out} will have an error, which will remain the same every time the input has the value 3. Mismatch, therefore, represents a fixed nonlinearity, causing harmonic distortion. Thus, for a sinewave input, the output spectrum will contain harmonics of the input sinewave. The ratio of the total harmonic distortion (THD) to the signal, THD/S, will be approximately the same as the RMS relative error E_{rms} of the C_i's. Thus, it is hard to obtain better than about 10-bit linearity performance with the converter.

The linearity may be improved by choosing the capacitors used in converting each input sample randomly, rather than deterministically, as described above [13]. Now the error will be in general different each time a

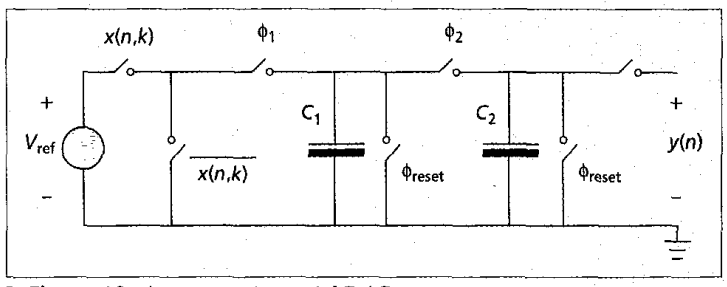


■ **Figure 8.** *Single-ended 2x residue amplifier: a) circuit diagram; b) circuit during phase 1; c) circuit during phase 2.*



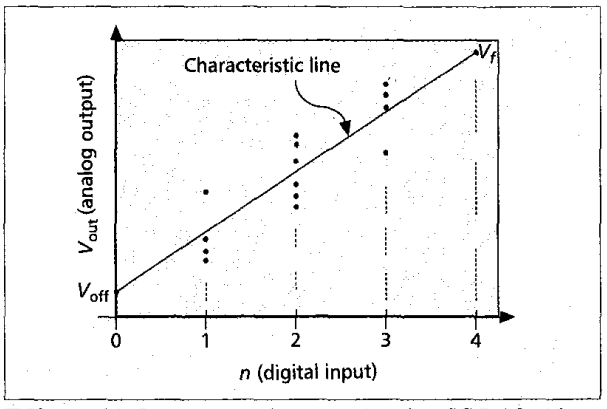
■ **Figure 9.** *Calibration process for a pipeline ADC.*

fixed code is entered into the DAC, and hence the matching errors introduce random noise, rather than distortion. It can be shown [13] that the ratio of the RMS noise to the full-scale signal is now $E_{rms}/2(MR)^{1/2}$, where M is the total number of the input capacitors C_i, and R is the oversampling ratio. Thus, using this strategy, the mismatch error is converted into a wideband noise, only a fraction of which falls in the signal band. This process can be regarded as *zero-order spectral shaping*.



■ **Figure 10.** *A two-capacitor serial DAC.*

First-order spectral shaping can be achieved based on the following considerations. Consider the input/output characteristics of a four-capacitor DAC (Fig. 11). For an input code of 0, the output is ideally 0 V; in reality, a small offset V_{off} will occur. For a full-scale input with a value 4, all capacitors are discharged into C_f, and ideally the output voltage will be $V_{out} = V_{ref}$. In fact, V_{out} will have a value V_f, which is slightly different from the reference voltage. V_f results from the combined contributions V_i of all input capacitors C_i, plus the fixed offset V_{off}. The line connecting these two end points thus has an offset error and a gain (slope) error. By themselves, these errors do not cause harmonic distortion, and are hence acceptable in most converter applications. If the points representing the three other intermediate input codes were to fall *exactly* on this characteristic line, therefore, the operation can be regarded as ideal. This condition cannot be achieved; however, it is possible to choose the elements used during conversion such that the *average* value of the output for each code falls on the characteristic line. This suppresses harmonic distortion and low-frequency errors.



■ **Figure 11.** *Input-output characteristics of an SC DAC with capacitor mismatch errors.*

To derive techniques for achieving this desirable situation, consider the point cluster corresponding to an input value $n = 1$ in Fig. 11. Each point corresponds to using a different C_i, and the ordinate of each point is $V_i + V_{off}$. Hence, the average value of output, if each capacitor is used equally often, is $\Sigma_i V_i/4 + V_{off}$. It can easily be shown that the point on the characteristic line for $n = 1$ has this same output value. The derivation can be repeated for the points representing the input values 2 and 3, with identical results. Hence, if all input elements are used with equal frequency for each code, then the average outputs will fall on a straight line, and hence the element mismatches will result in a noise-like error with a zero mean value. This indicates that the power spectral density (PSD) of the mismatch noise has a zero at dc, and hence the PSD is nonuniform. This process thus provides a first-order shaping of the mismatch noise.

There exist numerous techniques for achieving the required equal average usage for the individual capacitors. In one, called *barrel shifting*, the capacitors used for the first sample with value $m1$ are $C_1, C_2, \ldots, C_{m1}$; for the second sample with value $m2$, the set $C_2, C_3, \ldots, C_{m2+1}$ is used, etc., and the selection wraps around back to C_1 once the last of the C_i has been used. Another averaging technique, called *individual level averaging*, keeps track of the past usage of each element C_i for each input code, and assigns them so as to keep the average usage uniform. Yet another averaging technique, named *data-weighted averaging*, uses the set $C_1, C_2, \ldots, C_{m1}$ for the first sample, $C_{m1+1}, C_{m1+2}, \ldots, C_{m1+m2}$ for the second, etc., and wraps-around to C_1 after the last $C_i(C_M)$ has been used. These techniques have various relative advantages and disadvantages. The barrel-shifting method is simple to perform, but it does not guarantee equal usage of the capacitors. Also, for input frequencies related to the rotational frequency, it can generate undesirable tones in the passband. Individual level averaging, which is not so susceptible to tone generation, requires more elaborate digital circuitry, and takes longer to achieve the desired averaging. Data-weighted averaging is relatively simple, and achieves rapid averaging since no element will be used twice until all others are used. Other techniques have also been proposed for achieving first-order noise shaping. The reader is referred to Sec. 8.3.3 of [1] for a detailed discussion and journal paper references on these methods.

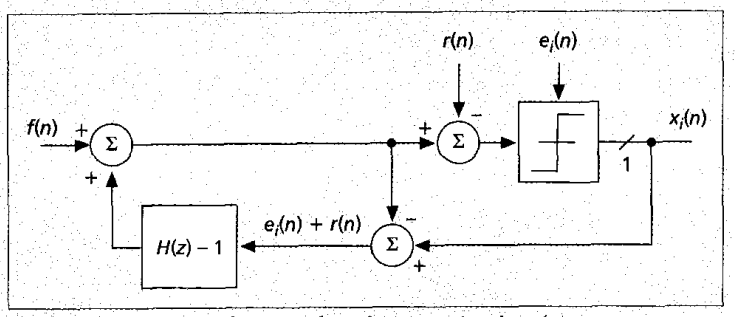
To achieve *higher-order noise shaping*, consider the error voltage dV_i present in the contribution V_i of the i-th capacitor C_i to the output voltage V_{out}. Since, as explained above, the maximum output V_f is accepted as being accurate, it follows that $\Sigma_i dV_i = 0$, and that the ideal contribution from each capacitor is $V_{id} = (V_f - V_{off})/M$. Assume now that the binary logic signal $x_i(n)$, which decides whether or not V_i will contribute to V_{out} in the n-th sampling period, can be written in the form

$$x_i(n) = f(n) + h(n) * [e_i(n) - r(n)] \qquad (2)$$

Here, the asterisk $*$ denotes the discrete-time convolution; also, $f(n)$ is a bounded function, independent of i, and $h(n)$ is the impulse response of the desired shaping filter. Finally, the $e_i(n)$ are pseudo-random bounded functions, in general different for each i. Then, the output error in the n-th period is

$$\begin{aligned} err(n) &= \sum_i x_i(n) dV_i \\ &= [f(n) - h(n) * r(n)] \sum_i dV_i + h(n) * \sum_i e_i(n) dV_i \end{aligned} \qquad (3)$$

The first term on the RHS is (by assumption) zero, and the second term contains the desired filter function. Hence, if we can generate a set of binary logic sequences x_i such that in each sampling period they satisfy Eq. 2 *and* their sum equals the input value $m(n)$, the error shaping is accomplished.



■ **Figure 12.** *Digital delta-sigma loop for generating the x_i(n) sequence of Eq. 2.*

Consider next the digital delta-sigma loop shown in Fig. 12 [14]. Analysis shows that its single-bit output sequence $x_i(n)$ is given exactly by (2), if the truncation error of the comparator is denoted by $e_i(n)$, and if $H(z)$ is the z-transform of $h(n)$. Hence, M such structures (one for each capacitor C_i) can be used to generate the $x_i(n)$ sequences for the operation of the DAC.

For a positive integer system, the common input $f(n)$ of the loops can be chosen so that the input of the truncation block in one of the loops is zero, and in all others it is positive. This will minimize the signals in the loops, and hence helps to keep their operation stable. The sequence $r(n)$ is essentially a time-variable threshold for the comparators. It is chosen such that exactly $m(n)$ of the M loops have outputs $x_i(n) = 1$ during period n. Note that this process can be combined with the correction technique illustrated in Fig. 2.

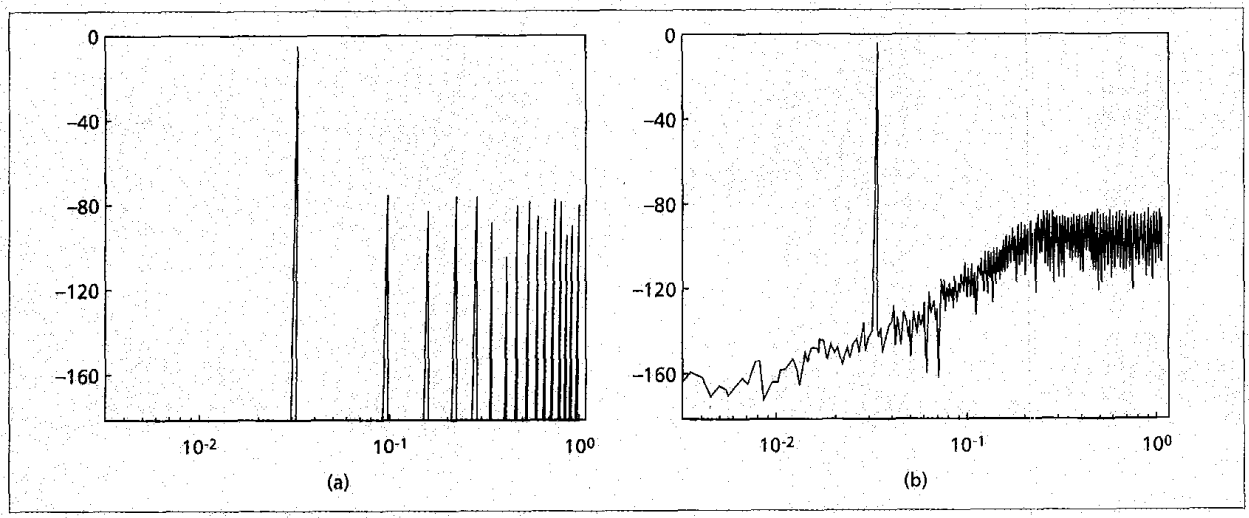
The mismatch error shaping process can be applied to other structures, such as the two-capacitor serial DAC described above and shown in Fig. 10 [15, 12]. Whereas in the case of the M-element DAC of Fig. 1 the degree of freedom that allowed spectral error shaping without changing the signal processing function was the arbitrary choice of the C_i in generating each analog output sample, here there is the option of interchanging the roles of C_1 and C_2 in each clock cycle when ϕ_1=1. As can be seen from the discussion given above including Eq. 1, interchanging C_1 and C_2 changes the sign of α, and hence also the sign of the error term in the sum that describes *Error* in Eq. 1. If the choice of the capacitors for the k-th bit of the n-th input word is described by the sequence $t(n,k) = \pm 1$, this sequence can be generated in such way that the resulting error has a shaped spectrum. The sequence generator is again a digital delta-sigma loop [15], which simulates the analog mismatch error generated by the actual DAC using the formula giving *Error* (without the unknown and unimportant constant factor α), and derives $t(n,k)$ in such a way that the power spectrum of *Error* is shaped. Figure 13 compares the unshaped and shaped output spectra of the DAC for a sinewave input with a peak-to-peak amplitude of $0.7V_{ref}$, an oversampling ratio of 10, and an assumed mismatch of α = 0.1 percent. Third-order noise shaping and dithering was used in the loop generating $t(n,k)$. The unshaped error gives a line spectrum with a S/THD ratio of only 70 dB; the S/(N+THD) for the mismatch-shaped spectrum is around 96 dB, a gain of 26 dB. Note that, unlike for the error-canceling scheme discussed earlier for this structure, mismatch shaping does not double (or change in any way) the conversion time; the only cost is the added digital circuitry, which is small.

CONCLUSIONS

In this tutorial , it was shown that very high accuracy and linearity may be obtained in data conversion even when using inaccurate analog components, by introducing additional digital logic that takes advantage of the hidden degrees of freedom in the operation of the converter circuit to achieve cancellation, calibration, or frequency shaping of the error introduced by the analog imperfections. This enables the designer of mixed-mode interface circuits to satisfy the increasing demands for ever faster and more accurate fully integrated data converters.

REFERENCES

[1] S. R. Norsworthy, R. Schreier, and G. C. Temes, *Delta-Sigma Data Converters*, Piscataway, NJ: IEEE Press, 1997.
[2] U. Moon, J. Steensgaard, and G. C. Temes, "A Switched-Capacitor DAC with Analog Mismatch Correction," to appear.
[3] H.J. Schouwenaars *et al.*, "An Oversampled Multibit CMOS D/A Converter for Digital Audio with 115-dB Dynamic Range," *IEEE J. Solid-State Circuits*, vol. 26, no. 12, Dec. 1991, pp. 1775–80.
[4] H. S. Lee, D. A. Hodges, and P. R. Gray, "A Self-Calibrating 15-bit CMOS



■ **Figure 13.** *Output spectra of the two-capacitor DAC of Fig. 10: a) unshaped; b) shaped.*

A/D Converter," *IEEE J. Solid-State Circuits*, vol. 19, no. 12, Dec. 1984, pp. 813–19.
[5] S. Lewis *et al.*, "A 10-bit 20-Msample/s Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 27, no. 3, Mar. 1992, pp. 351–58.
[6] H. Schmid, *Electronic Analog/Digital Conversion Techniques*, Van Nostrand, New York, 1970, pp. 323–25.
[7] S. Lewis and P. Gray, "A Pipelined 5-Msamples/s 9-bit Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. SC-22, Dec. 1987, pp. 954–61.
[8] A. Karanicolas, H. Lee, and K. Bacrania, "A 15-b 1-Msample/s Digitally Self-Calibrated Pipeline ADC," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, Dec. 1993, pp. 1207–15.
[9] G. C. Temes, "Autozeroing and Correlated Double Sampling Techniques," *Analog Circuit Design*, J. H. Huijsing, R. J. van de Plassche, and W. M. C. Sansen, Eds., Kluwer, 1996.
[10] R. Suarez, P. Gray, and D. Hodges, "ALL MOSFET Charge-Redistribution A/D Conversion Techniques–Part II," *IEEE J. Solid-State Circuits*, vol. 10, no. 12, Dec. 1975, pp. 379–85.
[11] P. Rombouts *et al.*, "Capacitor Mismatch Compensation for Quasi-Passive Switched-Capacitor DAC," *IEEE Trans. Circuits and Systems–I*, vol. 45, no. 1, Jan. 1998, pp. 68–71.
[12] J. Steensgaard, U. Moon, and G. C. Temes, "Mismatch-Shaped Pseudo-Passive Two-Capacitor DAC," *Proc. IEEE Alessandro Volta Memorial Wksp.*, Como, Italy, Mar. 1999.
[13] L. R. Carley, "A Noise-Shaping Coder Topology for 15+ Bit Converters," *IEEE J. Solid-State Circuits*, vol. 24, no. 4, Apr. 1989, pp. 267–73.
[14] R. Schreier and B. Zhang, "Noise-Shaped Multi-Bit D/A Converter Employing Unit Elements," *Elect. Lett.*, vol. 31, no. 20, Sept. 28, 1995, pp. 1712–13.
[15] J. Steensgaard, U. Moon, and G. C. Temes, "Mismatch-Shaping Serial Digital-to-Analog Converter," *IEEE Int'l. Symp. Circuits and Sys.*, vol. 2, Orlando, FL, June 1999, pp. 5–8.

BIOGRAPHIES

UN-KU MOON (moon@ece.orst.edu) received a Ph.D. in electrical engineering from the University of Illinois at Urbana-Champaign in 1994. From 1994 to 1998 he was a member of technical staff at Bell Laboratories, Lucent Technologies, and designed MOS telecommunication circuits. Since January 1998 he has been with Oregon State University. His research interest has been in the area of analog and mixed analog-digital integrated circuits. His current research interests include high-frequency switched-capacitor filters, low-voltage switched-capacitor circuits, high-resolution and oversampling data converters, and highly linear continuous-time filters.

JESPER STEENSGAARD (steensgaard@ieee.org) received a Ph.D. in electrical engineering from the Technical University of Denmark in 1999. From 1997 to 1998 he was a visiting researcher at Oregon State University. He is currently on the faculty at Columbia University, New York. His research has focused on sensor interfaces, low-voltage/low-power switched-capacitor circuits, and high-resolution data converters. Currently, he is developing high-speed delta-sigma A/D converters for digital-communication applications. He has several U.S. and international patents pending.

GABOR C. TEMES [LF] (temes@ece.orst.edu) received his Ph.D. in electrical engineering from the University of Ottawa, Canada. He held university positions at the Technical University of Budapest, Stanford University, and UCLA, and worked in industry at Bell-Northern Research and Ampex Corp. He is now a professor at Oregon State University. He received the Centennial Medal of the IEEE, as well as the Education Award and the Technical Achievement Award of the IEEE Circuits and Systems Society. He was the 1998 recipient of the IEEE Graduate Teaching Award. His recent research has dealt with CMOS analog integrated circuits, as well as data converters and integrated sensor interfaces.