

Why Dynamic-Element-Matching DACs Work

Ian Galton, *Member, IEEE*

Abstract—This jump-start tutorial brief explains the principle that underlies all of the published mismatch-scrambling and mismatch-shaping dynamic-element-matching (DEM) digital-to-analog converters (DACs). It explains the apparent paradox of how an all-digital algorithm can cause analog component mismatches to introduce spectrally shaped noise instead of nonlinear distortion, even though the algorithm has no knowledge of the actual mismatches. The concept is first explained in the context of a discrete-time three-level DEM DAC. The results are then generalized to an arbitrary number of levels, to segmented DEM DACs, and to continuous-time DEM DACs.

Index Terms—Digital-to-analog converter (DAC), dynamic element matching (DEM), mismatch scrambling, mismatch shaping.

I. INTRODUCTION

DYNAMIC-ELEMENT-MATCHING (DEM) digital-to-analog converters (DACs) have enabled major performance improvements over the last decade in oversampling delta-sigma data converters, pipelined ADCs, and high-resolution Nyquist-rate DACs [1]–[7]. In many cases, they effectively eliminate component mismatches as a performance-limiting source of error.

In DACs without DEM, mismatches among nominally identical circuit elements inevitably introduced during circuit fabrication cause nonlinear distortion. By scrambling the usage pattern of the elements, DEM causes the error resulting from the mismatches to be pseudorandom noise that is uncorrelated with the input sequence instead of nonlinear distortion.

In *mismatch-scrambling* DEM DACs, the noise is white, and in *mismatch-shaping* DEM DACs, the noise is spectrally shaped. The former are used in Nyquist-rate applications such as pipelined ADCs that require highly linear DACs. The latter are used in *oversampling* applications, i.e., applications such as delta-sigma ($\Delta\Sigma$) data converters in which the DAC's input signal occupies a bandwidth of much less than half the sample rate. The idea is to shape the power spectral density (PSD) of the noise to lie mostly outside of the input signal band.

A large variety of DEM DACs have been published with different properties and applications [8]–[20]. However, they all operate on the same underlying principle. The principle is counterintuitive because DEM is a completely digital technique that assumes no knowledge of the component mismatches, yet it controls the error introduced by the component mismatches.

This brief provides a tutorial explanation of the DEM principle in a framework that is not specific to any particular DEM

Manuscript received December 18, 2008. Current version published February 26, 2010. This paper was recommended by Associate Editor P. K. Hanumolu.

The author is with the Electrical and Computer Engineering Department, University of California, San Diego, La Jolla, CA, USA (e-mail: galton@ucsd.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSII.2010.2042131

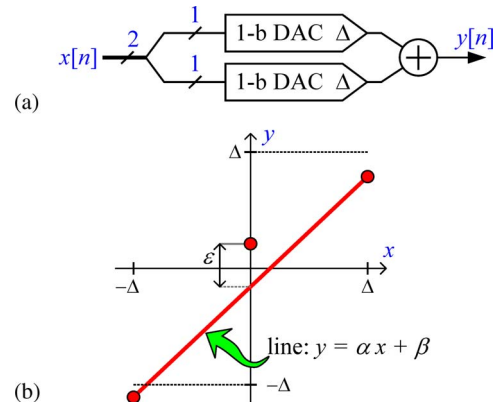


Fig. 1. (a) Three-level DAC with a thermometer coded input. (b) DAC's output values versus its three possible input values.

DAC variant. The basic DEM idea is described in Section II for the simplest possible case of a discrete-time three-level DEM DAC. The results are extended to DEM DACs with arbitrary numbers of levels in Section III, to segmented DEM DACs in Section IV, and to continuous-time DEM DACs in Section V.

II. SIMPLEST NONTRIVIAL CASE: THREE LEVELS

A. Ideal Behavior

The purpose of a DAC is to convert a sequence of input values $x[n]$, represented as a sequence of digital codewords, into an analog waveform $v(t)$. The input sequence is updated at times nT , where $n = 0, 1, 2, \dots$ is the sample number, and T is the sample period. In sampled-data applications such as switched-capacitor circuits, the DAC output $v(t)$ is sampled once per sample period, so the DAC can be viewed as having a discrete-time analog output $y[n] = v(nT)$ and can be analyzed as a purely discrete-time system. In the following, DEM is first explained in this discrete-time context. Then, the results are extended to the continuous-time case.

Consider a three-level discrete-time DAC with ideal output values $-\Delta$, 0 , and Δ , where Δ denotes the DAC's minimum *step size*. Suppose the DAC input $x[n]$ is represented as a 2-bit *thermometer code*. In this case, each codeword is 00, 01, or 11 and symbolizes the numerical value $-\Delta$, 0 , or Δ , respectively. In the absence of nonideal circuit behavior, the DAC would simply change the format of the data from digital to analog without introducing any numerical error, so $y[n] = x[n]$.

B. Nonlinearity From 1-bit DAC Mismatches

The three-level DAC can be implemented with two 1-bit DACs as shown in Fig. 1(a). The 1-bit DACs each generate analog output samples given by

$$y_i[n] = \begin{cases} \Delta/2 + e_{hi}, & \text{if } c_i[n] = 1 \\ -\Delta/2 + e_{li}, & \text{if } c_i[n] = 0 \end{cases} \quad (1)$$

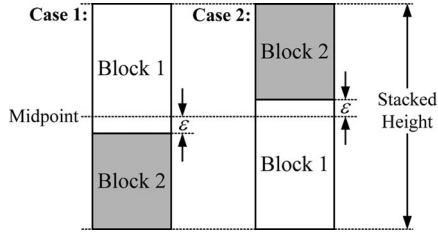


Fig. 2. Block stacking analogy.

for $i = 1$ and 2 , where e_{hi} and e_{li} are *mismatch errors* caused by mismatches among nominally identical circuit elements inevitably introduced during circuit fabrication. It is assumed that the mismatch errors do not significantly change over time. As indicated in Fig. 1(a), the input to each 1-bit DAC is one of the thermometer code input bits, and the two 1-bit DAC outputs are summed to obtain $y[n]$.

A plot of the three-level DAC's three output values versus its three possible input values is shown in Fig. 1(b) for a specific set of nonzero mismatch errors. A line is also shown that connects the points corresponding to $x[n] = \pm\Delta$. The line is described by the equation $y = \alpha x + \beta$, where α and β are the line's slope and offset, respectively. With this notation, Fig. 1(b) indicates that the three-level DAC's output can be written as

$$y[n] = \alpha x[n] + \beta + e_{\text{DAC}}[n] \quad (2)$$

where

$$e_{\text{DAC}}[n] = \begin{cases} \varepsilon, & \text{if } x[n] = 0, \\ 0, & \text{if } x[n] = -\Delta \text{ or } x[n] = \Delta. \end{cases} \quad (3)$$

It can be verified that the constants α , β , and ε depend only on the mismatch errors and that they reduce to 1, 0, and 0, respectively, for the ideal case of zero mismatch errors.

Thus, the three-level DAC can be viewed as introducing a constant *gain* of α , a constant *offset* of β , and an additive error term $e_{\text{DAC}}[n]$ that is a deterministic nonlinear function of $x[n]$. The nonlinearity is what prevents the three-level DAC's three output points from lying on a straight line. In typical applications, the DAC's gain and offset errors are rarely significant problems, but the nonlinear distortion introduced by $e_{\text{DAC}}[n]$ is often very problematic. The solution provided by DEM is to convert $e_{\text{DAC}}[n]$ from nonlinear distortion to either white or spectrally shaped pseudorandom noise that is uncorrelated with $x[n]$.

C. Conversion of Nonlinear Distortion Into Noise

It can be verified that if the input lines to the two 1-bit DACs are swapped, then α , β , ε , and (2) remain unchanged, and (3) changes only in that ε becomes $-\varepsilon$ [15]. This surprising symmetry result holds for any choice of mismatch errors. It is analogous to the less surprising result depicted in Fig. 2: two blocks of wood that differ in height by 2ε stack to the same total height regardless of the order in which they are stacked, and in each case, the seam between the two blocks is either ε above or ε below half the stacked height [21].

The DEM DAC shown in Fig. 3(a) exploits this result by sometimes swapping and sometimes not swapping the input bits to the two 1-bit DACs each time $x[n] = 0$. It is identical to the

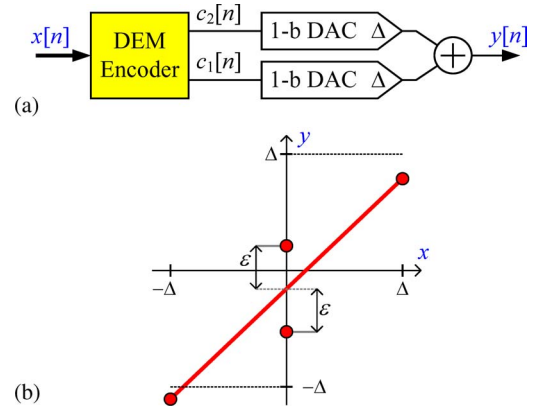


Fig. 3. (a) Three-level DEM DAC. (b) DAC's four output values versus its three possible input values.

three-level DAC in Fig. 1(a), except that it contains a digital *DEM encoder* prior to the 1-bit DACs. During each sample interval, the DEM encoder does one of two things: it either connects its top and bottom inputs to its top and bottom outputs, respectively, or connects its top and bottom inputs to its bottom and top outputs, respectively.

A plot of the three-level DEM DAC's four output values versus its three input values is shown in Fig. 3(b). Because of the symmetry result described above, each time $x[n] = 0$, the value of $y[n]$ is either ε above or ε below the line that joins the points corresponding to $x[n] = \pm\Delta$, depending on whether the DEM encoder swaps or does not swap the input bits to the 1-bit DACs.

It follows that the three-level DEM DAC satisfies (2) with $e_{\text{DAC}}[n] = s[n]\varepsilon$, where

$$s[n] = \begin{cases} 1, & \text{if } x[n] = 0, c_1[n] = 1, c_2[n] = 0 \\ -1, & \text{if } x[n] = 0, c_1[n] = 0, c_2[n] = 1 \\ 0, & \text{if } x[n] = -\Delta \text{ or } x[n] = \Delta. \end{cases} \quad (4)$$

Therefore, the sign of $e_{\text{DAC}}[n]$ is under the control of the DEM encoder. As before, the DAC output points do not lie on a straight line because $e_{\text{DAC}}[n]$ is nonzero when $x[n] = 0$. However, by pseudorandomly modulating the sign of $e_{\text{DAC}}[n]$, the DEM encoder can force $e_{\text{DAC}}[n]$ to be zero *on average*, which causes the *average* of the points to lie on the $y = \alpha x + \beta$ line. This causes the DEM DAC to be equivalent to a DAC with perfect linearity plus an additive zero-mean noise source.

In addition to ensuring that $e_{\text{DAC}}[n]$ is of zero mean, it would be desirable for $e_{\text{DAC}}[n]$ to be statistically independent of $x[n]$. However, it can be seen from (4) that this is not possible. For example, if the magnitude of $x[n]$ were always Δ , then $e_{\text{DAC}}[n]$ would always be zero, whereas if $x[n]$ were always zero, then $e_{\text{DAC}}[n]$ would always have a magnitude of unity.

Fortunately, it is possible to make $e_{\text{DAC}}[n]$ uncorrelated with $x[n]$, as demonstrated below. This is a requirement in most applications. Otherwise, the PSD of the DAC output would contain cross terms from the correlation of $x[n]$ and $e_{\text{DAC}}[n]$ that could cause spurious tones and other nonlinear artifacts.

D. Mismatch Shaping Versus Mismatch Scrambling

A *mismatch-scrambling* DEM DAC is one in which $e_{\text{DAC}}[n]$ is white noise. For the case of the three-level DEM DAC,

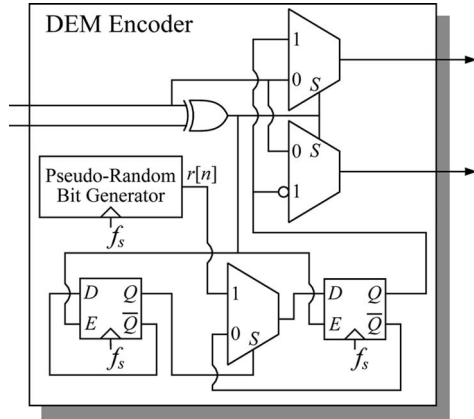


Fig. 4. First-order mismatch-shaping DEM encoder for the DEM DAC in Fig. 3(a) with a thermometer coded input.

mismatch scrambling can be achieved if the DEM encoder's swapping operation is controlled by a pseudorandom 1-bit sequence updated at the sample rate. Provided the 1-bit pseudorandom sequence is white, has an equal probability of being 1 or 0, and is independent of $x[n]$, it causes $s[n]$ and, therefore, $e_{\text{DAC}}[n]$ to be zero-mean white noise that is uncorrelated with $x[n]$, as required.

A *mismatch-shaping* DEM DAC is one in which $e_{\text{DAC}}[n]$ is spectrally shaped noise. A DEM encoder for a first-order mismatch-shaping version of the three-level DEM DAC is shown in Fig. 4 [22]. It consists of four 2–1 multiplexers, an XOR gate, two flip-flops, and a pseudorandom bit generator. The pseudorandom bit generator is designed such that $r[n]$ well approximates a white sequence of random variables that are uncorrelated with $x[n]$ and take on values of 1 and 0 with equal probability. Note that the data path is not clocked, so the only delays between the inputs and outputs of the DEM encoder are the logic propagation delays. The circuitry that determines whether $c_1[n] = 1$ and $c_2[n] = 0$ or vice versa when $x[n] = 0$ is clocked at the DAC's sample rate.

It can be verified from Fig. 4 and (4) that $s[n]$, $n = 0, 1, 2, \dots$, is a concatenation of the following two types of sequences:

$$1, 0, \dots, 0, -1, 0, \dots, 0 \quad \text{or} \quad -1, 0, \dots, 0, 1, 0, \dots, 0 \quad (5)$$

called Type 1 and Type 2 *doublet sequences*, respectively, where each 0 is present only when the corresponding value of $x[n]$ is nonzero [23]. For example, if $\{x[1], x[2], x[3], x[4], x[5], x[6], \dots\} = \{0, 0, -\Delta, 0, \Delta, 0, \dots\}$, $r[0] = 0$, and $r[3] = 1$, then $\{s[1], s[2], s[3], s[4], s[5], s[6], \dots\} = \{1, -1, 0, -1, 0, 1, \dots\}$, which is a Type 1 doublet followed by a Type 2 doublet.

The statistical properties of $r[n]$ ensure that the DEM encoder chooses the type of each doublet independently from its previous choices and $x[n]$ and such that the two doublet types occur with equal probability. Since $r[n]$ is independent of $x[n]$, $s[n]$ is uncorrelated with $x[n]$, as required. Furthermore, since the only nonzero values in each doublet in $s[n]$ are a single 1 and a single -1 , it follows that $s[n]$ has zero mean, as required, and also that

$$\left| \sum_{k=0}^n s[k] \right| \leq 1, \quad \text{for all } n > 0. \quad (6)$$

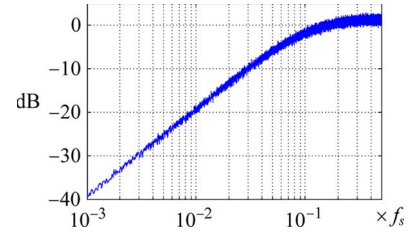


Fig. 5. Simulated PSD of $e_{\text{DAC}}[n]$ in decibels relative to ε^2 for the DEM DAC in Fig. 3(a) with the DEM encoder in Fig. 4.

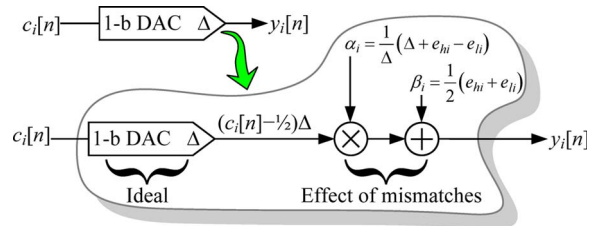


Fig. 6. One-bit DAC with mismatch errors that is equivalent to an ideal 1-bit DAC with constant gain and offset errors.

Given that $e_{\text{DAC}}[n] = s[n]\varepsilon$, this implies that

$$\lim_{N \rightarrow \infty} \frac{1}{N} \left| \sum_{k=0}^{N-1} e_{\text{DAC}}[k] \right|^2 = 0 \quad (7)$$

which indicates that the PSD of $e_{\text{DAC}}[n]$ is zero at $\omega = 0$. It can be shown that the PSD is free of spurious tones and rises at 20 dB/decade as ω increases from zero [24]. Hence, $e_{\text{DAC}}[n]$ has a first-order highpass shape, as illustrated in Fig. 5.

Highpass mismatch shaping of the type described above is useful when the DAC's input signal occupies a frequency band centered at $\omega = 0$. Other mismatch-shaping options are also available. In applications such as bandpass $\Delta\Sigma$ data converters, the DAC's input signal band is not centered at $\omega = 0$. In such cases, bandstop mismatch-shaping DACs can be used to suppress $e_{\text{DAC}}[n]$ in the signal band [17], [25]–[27]. In both cases, it is possible to achieve higher order spectral shaping [13], [16], [23].

III. EXTENSION TO MORE THAN THREE LEVELS

DEM works in the three-level DEM DAC because of the symmetry illustrated in Fig. 3(b). Similar symmetry is exploited by DEM DACs with more than three levels, but the corresponding graphical explanation is too complicated to easily provide insight. Therefore, a nongraphical explanation of DEM is developed in this section, first for the three-level DEM DAC and then for a DEM DAC with an arbitrary number of levels.

A comparison of (1) to the two possible output values of the 1-bit DAC shown in Fig. 6 indicates that mismatch errors in a 1-bit DAC are equivalent to gain and offset errors. Therefore, since the output of the three-level DEM DAC is $y[n] = y_1[n] + y_2[n]$, it can be written as

$$y[n] = \alpha_1 \left(c_1[n] - \frac{1}{2} \right) \Delta + \beta_1 + \alpha_2 \left(c_2[n] - \frac{1}{2} \right) \Delta + \beta_2. \quad (8)$$

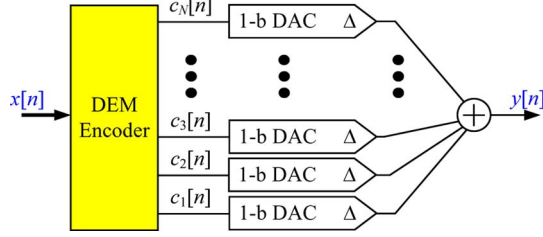


Fig. 7. $(N + 1)$ -level DEM DAC.

Since $c_1[n]$ and $c_2[n]$ are both 0 when $x[n] = -\Delta$ and both 1 when $x[n] = \Delta$, it follows from (4) that

$$c_1[n] = \frac{1}{2}(x[n]/\Delta + 1 + s[n]) \quad (9)$$

$$c_2[n] = \frac{1}{2}(x[n]/\Delta + 1 - s[n]). \quad (10)$$

Substituting (9) and (10) into (8) yields (2) with

$$\alpha = \frac{1}{2}(\alpha_1 + \alpha_2), \quad \beta = \beta_1 + \beta_2 \quad (11)$$

$$e_{\text{DAC}}[n] = \frac{\Delta}{2}(\alpha_1 - \alpha_2)s[n]. \quad (12)$$

This analysis shows that $e_{\text{DAC}}[n]$ is proportional to $s[n]$, as found previously. As shown below, the analysis easily extends to the case of DEM DACs with more than three levels.

A general $(N + 1)$ -level DEM DAC is shown in Fig. 7. It consists of a DEM encoder followed by N 1-bit DACs that operate according to (1). The overall DAC output is the sum of the 1-bit DAC outputs. The input sequence $x[n]$ is represented by a sequence of codewords that symbolize the set of values $\{-N\Delta/2, -N\Delta/2 + 1, -N\Delta/2 + 2, \dots, N\Delta/2\}$. The N 1-bit output sequences from the DEM encoder satisfy

$$c_1[n] + c_2[n] + \dots + c_N[n] = x[n]/\Delta + \frac{1}{2}N \quad (13)$$

which ensures that $y[n] = x[n]$ in the absence of mismatch errors.

The following two results can be obtained using the same argument that led to (9)–(12). First, each of the DEM encoder's 1-bit output sequences can be written as

$$c_i[n] = (m_i x[n] + \lambda_i[n]) / \Delta + \frac{1}{2} \quad (14)$$

where the m_i are constants, and $\lambda_i[n]$ are sequences that satisfy

$$\sum_{i=1}^N m_i = 1 \quad \text{and} \quad \sum_{i=1}^N \lambda_i[n] = 0, \quad (15)$$

respectively, for all n . Second, the $(N + 1)$ -level DEM DAC output is given by (2), where α and β are the average and sum of the 1-bit DAC α_i and β_i values, respectively, and

$$e_{\text{DAC}}[n] = \sum_{i=1}^N \alpha_i \lambda_i[n]. \quad (16)$$

A comparison of these results to (9)–(12) indicates that for the three-level DAC described previously, $m_1 = m_2 = 1/2$, $\lambda_1[n] = s[n]\Delta/2$, and $\lambda_2[n] = -s[n]\Delta/2$.

It follows from (14) that each 1-bit DEM encoder output is equal to $x[n]$ scaled by m_i/Δ , offset by $1/2$, and quantized to 0 or 1. The quantization error sequences are $\lambda_i[n]/\Delta$, $i = 1, 2, \dots, N$, and the right equation in (15) implies that they sum to zero, as required to satisfy (13). Mismatches among the α_i values of the 1-bit DACs cause this quantization error to leak into the overall DAC output via $e_{\text{DAC}}[n]$ according to (16).

Fortunately, the DEM encoder has considerable control over the $\lambda_i[n]$ sequences because, for most values of $x[n]$, there are multiple solutions to (13) and the different solutions result in different $\lambda_i[n]$ values. DEM encoders are designed to exercise this control such that the $\lambda_i[n]$ sequences have the statistical properties desired of $e_{\text{DAC}}[n]$. For example, a DEM encoder for an $(N + 1)$ -level first-order mismatch-shaping DEM DAC can be implemented by wiring together N copies of the DEM encoder shown in Fig. 4 in the form of a tree structure [22].

IV. EXTENSION TO SEGMENTED DACS

The DEM DAC architecture shown in Fig. 7 has the disadvantage that the DEM encoder's complexity and the number of 1-bit DACs grow exponentially with the number of bits of DAC resolution. Hence, it is not practical in applications such as high-resolution Nyquist-rate DACs.

Recently, *segmented* DEM DACs have been developed that overcome this problem [7], [28], [29]. Segmented DEM DACs have the same general structure shown in Fig. 7, except that the 1-bit DAC weights are not all equal and their DEM encoders function somewhat differently. By having groups of 1-bit DACs with equal weights yet having the weights of the 1-bit DACs in each group be larger than those of the previous group, sufficient flexibility can be retained for DEM to be effective without having complexity that grows exponentially with the number of input bits [7], [29].

In a segmented DEM DAC, each 1-bit DAC operates according to

$$y_i[n] = \begin{cases} K_i \Delta/2 + e_{hi}, & \text{if } c_i[n] = 1 \\ -K_i \Delta/2 + e_{li}, & \text{if } c_i[n] = 0 \end{cases} \quad (17)$$

where K_i is the *weight* of the 1-bit DAC. By definition, $K_1 = 1$, and each K_i , for $i = 2, 3, \dots, N$, is a positive integer ordered such that $K_i \geq K_{i-1}$. The values of K_i and the DEM encoder are designed such that

$$K_1 c_1[n] + K_2 c_2[n] + \dots + K_N c_N[n] = x[n]/\Delta + \frac{1}{2}M \quad (18)$$

where $M = K_1 + K_2 + \dots + K_M$. This ensures that $y[n] = x[n]$ in the absence of mismatch errors where each sample of $x[n]$ can be any value in the set

$$\left\{ -\frac{1}{2}M\Delta, -\left(\frac{1}{2}M - 1\right)\Delta, -\left(\frac{1}{2}M - 2\right)\Delta, \dots, \frac{1}{2}M\Delta \right\}. \quad (19)$$

An argument almost identical to that of the previous section indicates that the DEM encoder's 1-bit output sequences are still given by (14), except that m_i and $\lambda_i[n]$ satisfy

$$\sum_{i=1}^N K_i m_i = 1 \quad \text{and} \quad \sum_{i=1}^N K_i \lambda_i[n] = 0, \quad (20)$$

respectively, and (2) still holds but with

$$e_{\text{DAC}}[n] = \sum_{i=1}^N \alpha_i K_i \lambda_i[n]. \quad (21)$$

Therefore, in analogy to the nonsegmented case, the DEM encoder in a segmented DEM DAC is designed to exercise the flexibility it has in satisfying (18) such that the $\lambda_i[n]$ sequences have the statistical properties desired of $e_{\text{DAC}}[n]$.

Segmented DEM DACs offer huge reductions in circuit area relative to their nonsegmented counterparts when a large number of input levels are required. However, this benefit comes at a price. As shown in [29], it is only possible for the DEM encoder to ensure that $e_{\text{DAC}}[n]$ does not have a nonlinear dependence on $x[n]$ if the range of $x[n]$ is restricted so that it never takes on the smallest $K_N - 1$ values or the largest $K_N - 1$ values in (19).

For example, the DEM DAC with the highest level of segmentation published to date can handle an input sequence that takes on any of 32 767 values, but it is necessary to restrict the input sequence to the middle 16 385 values of this range to ensure that the error resulting from mismatches is free of nonlinear distortion [30]. This corresponds to a 6-dB reduction in signal swing. In terms of the signal-to-noise ratio, the signal-swing reduction can be compensated by reducing the circuit noise from the 1-bit DACs by 6 dB, but doing so usually dictates a significant increase in power dissipation. Therefore, segmentation represents a fundamental tradeoff between circuit complexity and power dissipation.

V. EXTENSION TO CONTINUOUS-TIME DACS

In cases where the DEM DAC output is not sampled by subsequent circuitry, the DAC output must be treated as a continuous-time waveform. Fortunately, all the DEM results described above hold with a few simple modifications, as described below.

The same DEM DAC architecture as described previously is used for the continuous-time case, except that, during the n th sample period, i.e., when $nT \leq t < (n+1)T$, the output of each 1-bit DAC is given by

$$y_i(t) = \begin{cases} a(t-nT)K_i\Delta/2 + e_{hi}(t-nT), & \text{if } c_i[n] = 1 \\ -a(t-nT)K_i\Delta/2 + e_{li}(t-nT), & \text{if } c_i[n] = 0 \end{cases} \quad (22)$$

where $a(t)$ is the ideal unit output pulse common to all of the 1-bit DACs, and $e_{hi}(t)$ and $e_{li}(t)$ are *mismatch error pulses* caused by the component mismatches and are specific to the i th 1-bit DAC. The only assumption made about $a(t)$, $e_{hi}(t)$, and $e_{li}(t)$ is that they are zero outside of $0 \leq t < T$.

An equivalent form of (22) is

$$y_i(t) = \alpha_i(t-nT)K_i \left(c_i[n] - \frac{1}{2} \right) \Delta + \beta_i(t-nT) \quad (23)$$

where

$$\alpha_i(t) = a(t) + \frac{e_{hi}(t) - e_{li}(t)}{K_i\Delta} \quad \beta_i(t) = \frac{e_{hi}(t) + e_{li}(t)}{2}. \quad (24)$$

This is the continuous-time counterpart of the result shown in Fig. 6. It can be verified by substituting (24) into (23) to obtain (22).

As shown in [7], during the n th sample period

$$y(t) = \alpha(t-nT)x[n] + \beta(t-nT) + e_{\text{DAC}}(t) \quad (25)$$

where $\alpha(t)$ and $\beta(t)$ are pulses that are zero outside of $0 \leq t < T$, and

$$e_{\text{DAC}}(t) = \sum_{i=1}^N K_i \lambda_i[n] \alpha_i(t-nT). \quad (26)$$

The only differences between (25) and (26) and the corresponding results for the discrete-time case, i.e., (2) and (21), are that the constant coefficients α , β , and α_i in (2) and (21) are replaced by corresponding time pulses in (25) and (26). Consequently, all of the results described previously for the discrete-time case hold for the continuous-time case too. This implies that DEM applies to not only static mismatch errors but also pulse shape and timing mismatch errors.

REFERENCES

- [1] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. Hoboken, NJ: Wiley, 2005.
- [2] B. Jewett, K. Poulton, K.-C. Hsieh, and J. Doernberg, "A 12b 128 MSample/s ADC with 0.05LSB DNL," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 1997, pp. 138–139.
- [3] P. Rombouts and L. Weyten, "Dynamic element matching for pipelined A/D conversion," in *Proc. IEEE Int. Conf. Electron., Circuits Syst.*, Sep. 1998, vol. 2, pp. 315–318.
- [4] I. Galton, "Digital cancellation of D/A converter noise in pipelined A/D converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 47, no. 3, pp. 185–196, Mar. 2000.
- [5] K. O. Sullivan, C. Gorman, M. Hennessy, and V. Callaghan, "A 12-bit 320-Msample/s current-steering CMOS D/A converter in 0.44 mm²," *IEEE J. Solid-State Circuits*, vol. 39, no. 7, pp. 1064–1072, Jul. 2004.
- [6] B. Jewett, J. Liu, and K. Poulton, "A 1.2 Gs/s 15b DAC for precision signal generation," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2005, pp. 110–111.
- [7] K. L. Chan, J. Zhu, and I. Galton, "Dynamic element matching to prevent nonlinear distortion from pulse-shape mismatches in high-resolution DACs," *IEEE J. Solid-State Circuits*, vol. 43, no. 9, pp. 2067–2078, Sep. 2008.
- [8] L. R. Carley, "A noise-shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 267–273, Apr. 1989.
- [9] W. Redman-White and D. J. L. Bourner, "Improved dynamic linearity in multi-level $\Sigma\Delta$ converters by spectral dispersion of D/A distortion products," in *Proc. IEEE Eur. Conf. Circuit Theory Des.*, Sep. 1989, pp. 205–208.
- [10] B. H. Leung and S. Sutarja, "Multi-bit $\Sigma - \Delta$ A/D converter incorporating a novel class of dynamic element shaping," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 39, no. 1, pp. 35–51, Jan. 1992.
- [11] M. J. Story, "Digital to analogue converter adapted to select input sources based on a preselected algorithm once per cycle of a sampling signal," U.S. Patent 5 138 317, Aug. 11, 1992.
- [12] H. S. Jackson, "Circuit and method of cancelling nonlinearity error associated with component mismatches in a data converter," U.S. Patent 5 221 926, Jun. 22, 1993.
- [13] R. Schreier and B. Zhang, "Noise-shaped multibit D/A converter employing unit elements," *Electron. Lett.*, vol. 31, no. 20, pp. 1712–1713, Sep. 1995.
- [14] R. T. Baird and T. S. Fiez, "Linearity enhancement of $\Delta\Sigma$ A/D and D/A converters using data weighted averaging," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 42, no. 12, pp. 753–762, Dec. 1995.

- [15] T. W. Kwan, R. W. Adams, and R. Libert, "A stereo multibit Sigma Delta DAC with asynchronous master-clock interface," *IEEE J. Solid-State Circuits*, vol. 31, no. 12, pp. 1881–1887, Dec. 1996.
- [16] I. Galton, "Spectral shaping of circuit errors in digital-to-analog converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 10, pp. 808–817, Oct. 1997.
- [17] T. Shui, R. Schreier, and F. Hudson, "Mismatch shaping for a current-mode multibit delta-sigma DAC," *IEEE J. Solid-State Circuits*, vol. 34, no. 3, pp. 331–338, Mar. 1999.
- [18] I. Fujimori, L. Longo, A. Hairapetian, K. Seiyama, S. Kasic, J. Cao, and S. Chan, "A 90 dB SNR 2.5 MHz output rate ADC using cascaded multibit $\Delta\Sigma$ modulation at $8\times$ oversampling ratio," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2000, pp. 338–339.
- [19] A. A. Hamoui and K. W. Martin, "High-order multibit modulators and pseudo data-weighted-averaging in low-oversampling $\Delta\Sigma$ ADCs for broad-band applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 1, pp. 72–85, Jan. 2004.
- [20] K. Nguyen, A. Bandyopadhyay, R. Adams, K. Sweetland, and P. Baginski, "A 108 dB SNR 1.1 mW oversampling audio DAC with a three-level DEM technique," *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2592–2600, Dec. 2008.
- [21] J. K. Poulsen, private communication.
- [22] E. Fogleman, I. Galton, W. Huff, and H. Jensen, "A 3.3-V single-poly CMOS audio ADC delta-sigma modulator with 98-dB peak SINAD and 105-dB peak SFDR," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 297–307, Mar. 2000.
- [23] J. Welz and I. Galton, "Simplified logic for first-order and second-order mismatch-shaping digital-to-analog converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 48, no. 11, pp. 1014–1027, Nov. 2001.
- [24] J. Welz and I. Galton, "A tight signal-band power bound on mismatch noise in a mismatch-shaping digital-to-analog converter," *IEEE Trans. Inf. Theory*, vol. 50, no. 4, pp. 593–607, Apr. 2004.
- [25] H. Lin and R. Schreier, "A bandpass mismatch-shaped multi-bit $\Sigma\Delta$ switched-capacitor DAC using butterfly shuffler," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 1999, pp. 58–59.
- [26] T. S. Kaplan, J. F. Jensen, C. H. Fields, and M. F. Chang, "A 2-GS/s 3-bit $\Sigma\Delta$ -modulated DAC with tunable bandpass mismatch shaping," *IEEE J. Solid-State Circuits*, vol. 40, no. 3, pp. 603–610, Mar. 2005.
- [27] S. Reekmans, J. De Maeyer, P. Rombouts, and L. Weyten, "Quadrature mismatch shaping for digital-to-analog converters," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 12, pp. 2529–2538, Dec. 2006.
- [28] R. Adams, K. Q. Nguyen, and K. Sweetland, "A 113-dB SNR oversampling DAC with segmented noise-shaped scrambling," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1871–1878, Dec. 1998.
- [29] K. L. Chan, N. Rakuljic, and I. Galton, "Segmented dynamic element matching for high-resolution digital-to-analog conversion," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 11, pp. 3383–3392, Dec. 2008.
- [30] K. L. Chan and I. Galton, "A 14b 100 MS/s DAC with fully segmented dynamic element matching," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2006, pp. 2390–2399.