Necessary and Sufficient Conditions for Mismatch Shaping in a General Class of Multibit DACs

Jared Welz, *Member, IEEE,* and Ian Galton, *Member, IEEE*

***Abstract*—Multibit digital-to-analog converters (DACs) are often constructed by combining several 1-bit DACs of equal or different weights in parallel. In such DACs, component mismatches give rise to signal dependent error that can be viewed as additive *DAC noise*. In some cases these DACs use dynamic element matching techniques to decorrelate the DAC mismatch noise from the input sequence and suppress its power in certain frequency bands. Such DACs are referred to as mismatch-shaping DACs and have been used widely as enabling components in state-of-the-art $\Delta\Sigma$ data converters. Several different mismatch-shaping DAC topologies have been presented, but theoretical analyses have been scarce and no general unifying theory has been presented in the previously published literature. This paper presents such a unifying theory in the form of necessary and sufficient conditions for a multibit DAC to be a mismatch-shaping DAC and applies the conditions to evaluate the DAC noise generated by several of the previously published mismatch-shaping DACs and qualitatively compare their behavior.**

***Index Terms*—Analog-to-digital (A/D), data converters, $\Delta\Sigma$, digital-to-analog (D/A), dynamic element matching, linearized digital-to-analog converters (DACs), mismatch shaping, $\Sigma\Delta$, spectral shaping.**

I. Introduction

MOST multibit digital-to-analog converters (DACs) consist of multiple 1-bit DACs. In each case, the digital input sequence is decomposed into multiple 1-bit sequences each of which drives a 1-bit DAC. Each 1-bit DAC generates one of two analog output levels depending upon whether its input bit is high or low. The outputs of the 1-bit DACs are summed to form the output of the multibit DAC. The primary differences among the various multibit DAC architectures reside in how the multibit input sequence is mapped to the multiple 1-bit DAC input sequences, and how the output levels of the 1-bit DACs are scaled relative to each other.

Manuscript received March 24, 2002; revised November 12, 2002. The review process for this paper was administered by Associate Editor G. Temes, independently from the Editor-in-Chief and other editorial staff of this TRANSACTIONS. This work was supported in part by the University of California Communications Research Program under Grant Core 00-10069 and in part by the National Science Foundation under Grant CCR-0073552.

J. Welz was with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093-0407 USA. He is now with the Custom IC Design Group, Northrop Grumman Space Technology, Redondo Beach, CA 90278 USA (e-mail: jwelz@ece.ucsd.edu).

I. Galton is with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093-0407 USA (e-mail: galton@ece.ucsd.edu).

Digital Object Identifier 10.1109/TCSII.2002.807269

In practice, component mismatches inevitably introduced during circuit fabrication, most notably mismatches among nominally identical unit capacitors or current sources, cause the 1-bit DAC output levels to deviate from their ideal values. The resulting error can be modeled, without approximation, as additive error and is referred to as *DAC noise*. In present VLSI technology, the values of nominally identical components can rarely be matched to better than a standard deviation of 0.1%. In Nyquist-rate DACs, i.e., DACs that convert digital signals with a passband from zero up to half their sample-rate, this translates into DAC noise that limits the achievable signal-to-noise-and-distortion ratio (SINAD) to less than 70 dB. Moreover, without some form of dither or other randomization technique, the DAC noise is a deterministic, nonlinear function of the input sequence so it contains harmonic distortion which can be problematic in many applications.

Dynamic element matching (DEM) techniques can be applied to multibit DACs both to suppress the power of the DAC mismatch noise in specific frequency bands and to eliminate the harmonic distortion. Such multibit DACs are referred to as *mismatch-shaping DACs*. They are particularly useful in applications that require high precision within relatively narrow frequency bands. As such, in recent years they have become widely used in high-performance delta–sigma ($\Delta\Sigma$) data converters.

Although numerous mismatch-shaping DAC architectures have been developed, published mathematical analyses of these DACs have been limited and disjoint to date. Most analyses have been individually tailored to specific architectures, and in most cases simulations have been relied upon to determine the characteristics of the DAC noise, which can be misleading. Consequently, there is no unifying theory that applies to multibit DACs in general. This lack of theory has made it difficult to compare the merits of the different mismatch-shaping DAC architectures, and likely has impeded the development of new mismatch-shaping DAC architectures.

This paper provides a unifying theory in the form of necessary and sufficient conditions for a general multibit DAC to be a mismatch-shaping DAC. Unlike previous analyses [1] the conditions do not rely on properties of the component mismatches. The utility of the conditions is demonstrated by using them to analyze and qualitatively compare most of the widely used mismatch-shaping DAC architectures published to date: first-order, lowpass implementations of the vector feedback [2], data-weighted averaging (DWA) [3], [4], butterfly shuffler [5], tree structured [6], segmented butterfly shuffler [7], and partitioned DWA [8] DACs.

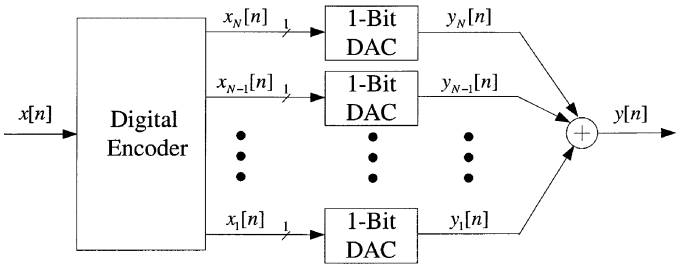


Fig. 1. The general multibit DAC.

II. The General Multibit DAC

The general multibit DAC shown in Fig. 1 consists of a digital encoder and a bank of N 1-bit DACs. The output of the ith 1-bit DAC is given by

$$y_i[n] = \begin{cases} \dfrac{\Delta_i}{2} + e_{h_i}, & \text{if } x_i[n] \text{ is high} \\ -\dfrac{\Delta_i}{2} + e_{l_i}, & \text{if } x_i[n] \text{ is low} \end{cases} \tag{1}$$

where Δ_i is the nominal step size of the ith 1-bit DAC, and e_{h_i} and e_{l_i} are its high and low errors, respectively. In many applications, the 1-bit DAC errors result from component mismatches introduced during fabrication of the 1-bit DACs. As such, they are modeled here as arbitrary constants. The digital encoder output is a vector, $\vec{x}[n]$, of N 1-bit sequences, $x_1[n], \ldots, x_N[n]$. The value of each 1-bit sequence is taken to be 1/2 when it is high and $-1/2$ when it is low. Ideally, a DAC's output is a scaled version of its input. To ensure that the multibit DAC approaches this ideal behavior when the 1-bit DAC errors approach zero, the digital encoder determines its output sequences under the following restriction:

$$\sum_{i=1}^{N} \Delta_i \cdot x_i[n] = \Delta_D \cdot x[n] \tag{2}$$

where Δ_D is the nominal smallest step size of the multibit DAC. Thus, if the 1-bit DAC errors were all zero, (1) and (2) imply that the DAC output would be given by

$$y[n] = \Delta_D x[n]. \tag{3}$$

However, in practice the 1-bit DAC errors are nonzero, and, as a result, the multibit DAC output is a nonlinear function of the multibit DAC input. The error from this nonlinearity can be written as additive error

$$y[n] = \Delta_D x[n] + \tilde{e}[n]. \tag{4}$$

The error sequence $\tilde{e}[n]$ often contains a constant offset and scaled version of the input; therefore, it is convenient to write (4) as

$$y[n] = \alpha x[n] + \beta + e[n] \tag{5}$$

where α and β are constants, and $e[n]$ is called the *DAC noise*. In a well-designed system, the DAC noise is a zero mean sequence that is uncorrelated from the multibit DAC input, and the constants α and β depend only on the 1-bit DAC errors.

Mismatch-shaping DACs are designed such that the digital encoder has several possible output vector values, $\vec{x}[n]$, that satisfy (2) for most DAC input values. For example, in a multibit DAC in which all the 1-bit DACs have the same nominal step size, a nominal output value of zero is obtained for any output vector with an equal number of high and low bit values. By exploiting this flexibility, the DAC noise can be tailored so that its power spectral density (PSD) has desired properties regardless of the values of the 1-bit DAC errors. This leads to the following definition for mismatch shaping.

Definition: A multibit DAC is said to *produce DAC noise with a given set of PSD properties* if, for any DAC input and collection of 1-bit DAC errors, there exist constants α and β, and a sequence $e[n]$ with the given set of PSD properties such that $y[n] = \alpha x[n] + \beta + e[n]$.

Various DAC noise PSD properties can be obtained by mismatch-shaping DACs. In some DACs, the digital encoder operates such that the DAC noise is white; i.e., its PSD is constant with respect to frequency. In such DACs, the power of the white noise depends upon the 1-bit DAC errors (e.g., larger 1-bit DAC errors tend to increase the power of the DAC noise), but the DAC noise is white for any choice of the 1-bit DAC errors. In other DACs, the digital encoder operates such that the DAC noise PSD is continuous with a value of zero at zero frequency: $\omega = 0$. In such cases, the power of the DAC noise tends to reside predominantly at high frequencies. Again, the overall power of the DAC noise depends upon the 1-bit DAC errors, but the zero at $\omega = 0$ and the weighting of the PSD toward high frequencies occurs for any choice of 1-bit DAC errors. Various other DACs are possible that achieve different DAC noise properties. In each case, specific properties (e.g., zero location) of the DAC noise PSD are preserved regardless of the 1-bit DAC errors.

Most mismatch-shaping DACs known to the authors adhere to the general DAC architecture shown in Fig. 1. The results presented in this paper apply to this class of DACs with the definition for mismatch shaping provided above. However, it should be noted that there exist mismatch-shaping DACs which do not adhere to this general architecture (e.g., see [9] and [10]), and the results presented in the paper are not directly applicable to them.

III. Conditions for Mismatch Shaping

The theorem below presents a necessary and sufficient condition for the general multibit DAC to produce DAC noise with a given set of PSD properties.

Theorem: The multibit DAC in Fig. 1 produces DAC noise with a given set of PSD properties if and only if there exist $N-1$ sequences $\phi_1[n], \ldots, \phi_{N-1}[n]$ such that: a) each digital encoder output is given by

$$x_i[n] = m_i x[n] + \sum_{j=1}^{N-1} d_{i,j} \cdot \phi_j[n] \tag{6}$$

where $d_{i,j}$ and m_i are constants, and b) for any selection of the $N-1$ constants $c_1, \ldots, c_{N-1}$, there exist two constants a and

b, and a sequence $\varepsilon[n]$ with the given set of PSD properties such that

$$\sum_{j=1}^{N-1} c_j \cdot \phi_j[n] = ax[n] + b + \varepsilon[n]. \tag{7}$$

Proof: Because $x_i[n]$ is interpreted as 1/2 when high and $-1/2$ when low, (1) can be written as

$$y_i[n] = \xi_i x_i[n] + \gamma_i \tag{8}$$

where $\xi_i \equiv \Delta_i + (e_{h_i} - e_{l_i})$ and $\gamma_i \equiv (e_{h_i} + e_{l_i})/2$. Given $y[n] = \sum_{i=1}^{N} y_i[n]$, (8) implies that

$$y[n] = \sum_{i=1}^{N} \xi_i x_i[n] + \beta_o \tag{9}$$

where $\beta_o \equiv \sum_{i=1}^{N} \gamma_i$.

Sufficiency: Assume that the $N-1$ sequences, $\phi_1[n], \ldots, \phi_{N-1}[n]$, exist and satisfy a) and b) in the theorem. Substituting (6) into (9) gives

$$y[n] = \underbrace{\left(\sum_{i=1}^{N} \xi_i m_i\right)}_{\equiv \alpha_o} x[n] + \sum_{j=1}^{N-1} \underbrace{\left(\sum_{k=1}^{N} \xi_k \, d_{k,j}\right)}_{\equiv c_j} \phi_j[n] + \beta_o. \tag{10}$$

Condition b) implies that the second summation in (10) can be decomposed as in (7). Thus, substituting (7) into (10) gives

$$y[n] = \underbrace{(a + \alpha_o)}_{\equiv \alpha} x[n] + \underbrace{b + \beta_o}_{\equiv \beta} + \varepsilon[n] \tag{11}$$

where $\varepsilon[n]$ has the given set of PSD properties, so the multibit DAC produces DAC noise with the given set of PSD properties.

Necessity: Let A be an $N \times N$ invertible matrix whose values are denoted $a_{j,k}$ (where j and k are the row and column numbers, respectively) and whose Nth row satisfies $a_{N,k} \equiv \Delta_k/\Delta_D$ for each k. Furthermore, for $j = 1, \ldots, N$, let

$$\phi_j[n] \equiv \sum_{k=1}^{N} a_{j,k} \cdot x_k[n]. \tag{12}$$

Using matrix notation, (12) can be written as $\vec{\phi}[n] = A\vec{x}[n]$. Because $a_{N,k} \equiv \Delta_k/\Delta_D$ for each k, (2) and (12) imply that $\phi_N[n] = x[n]$. Let D, whose value in its ith row and jth column is denoted $d_{i,j}$, be the inverse matrix of A. This implies that $\vec{x}[n] = D\vec{\phi}[n]$ and, for each i

$$x_i[n] = \sum_{j=1}^{N} d_{i,j} \cdot \phi_j[n]. \tag{13}$$

With $m_i \equiv d_{i,N}$, (13) is identical to (6) because $\phi_N[n] = x[n]$. Therefore, the $N-1$ sequences $\phi_1[n], \ldots, \phi_{N-1}[n]$ satisfy condition a) in the theorem.

To show that the $N-1$ sequences satisfy condition b) in the theorem, assume the multibit DAC produces DAC noise with the given set of PSD properties. In (9), ξ_i and β_o are arbitrary constants because each DAC error is an arbitrary constant. Thus, by assumption, *for any* selection of the constants $\xi_1, \ldots, \xi_N$, and β_o, there exist constants a and b, and a sequence $\varepsilon[n]$ with the given set of PSD properties such that

$$\sum_{i=1}^{N} \xi_i x_i[n] + \beta_o = ax[n] + b + \varepsilon[n]. \tag{14}$$

It follows from (12) that

$$\sum_{j=1}^{N-1} c_j \cdot \phi_j[n] = \sum_{i=1}^{N} \underbrace{\left(\sum_{j=1}^{N-1} c_j \cdot a_{j,i}\right)}_{\equiv d_i} x_i[n] \tag{15}$$

for any selection of $N-1$ constants, $c_1, \ldots, c_{N-1}$. Since (14) is satisfied for any selection of ξ_i and β_o, suppose $\xi_i = d_i$ for each i, and $\beta_o = 0$. In this case, the left-hand side of (14) is the same as the right-hand side of (15), which implies (7). Thus, the $N-1$ sequences satisfy condition b) in the theorem. ■

Therefore, in mismatch-shaping DACs, there are $N-1$ underlying sequences that, given the DAC input, determine the digital encoder outputs and, when linearly combined, produce a sequence that has the same form as the DAC output, i.e., $ax[n] + b + \varepsilon[n]$, where the gain and offset depend on the coefficients in this linear combination, and the sequence $\varepsilon[n]$ has the same PSD properties as the DAC noise.

The theorem can be used to show that the DAC noise from a given architecture has certain PSD properties. However, the corollary presented next is more convenient for this application.

Corollary 1: Given the multibit DAC shown in Fig. 1, let A be an $N \times N$ invertible matrix whose values are denoted $a_{j,k}$ (where j and k are the row and column numbers, respectively) and whose Nth row satisfies $a_{N,k} \equiv \Delta_k/\Delta_D$. Then, given

$$\phi_j[n] \equiv \sum_{k=1}^{N} a_{j,k} \cdot x_k[n] \tag{16}$$

for $j = 1, \ldots, N-1$, the multibit DAC produces DAC noise with a given set of PSD properties if and only if, for any selection of the $N-1$ constants $c_1, \ldots, c_{N-1}$, there exist two constants a and b, and a sequence $\varepsilon[n]$ with the given set of PSD properties such that

$$\sum_{j=1}^{N-1} c_j \cdot \phi_j[n] = ax[n] + b + \varepsilon[n]. \tag{17}$$

Proof: The proof follows directly from that of the theorem as the $N-1$ sequences in the corollary are formed the same way as in the proof of the theorem. ■

Therefore, to show that the DAC noise PSD from a given multibit DAC has a certain property, derive the $N-1$ sequences, $\phi_1[n], \ldots, \phi_{N-1}[n]$, as described in the corollary and show that any linear combination of these sequences can be written as in (17). The $N-1$ sequences in the corollary result from linear combinations of the digital encoder outputs, and there are many possible choices for these sequences. However, for a given multibit DAC, these sequences can often be chosen to minimize the effort required to show they satisfy (17). Several examples of this application are presented in the following section.

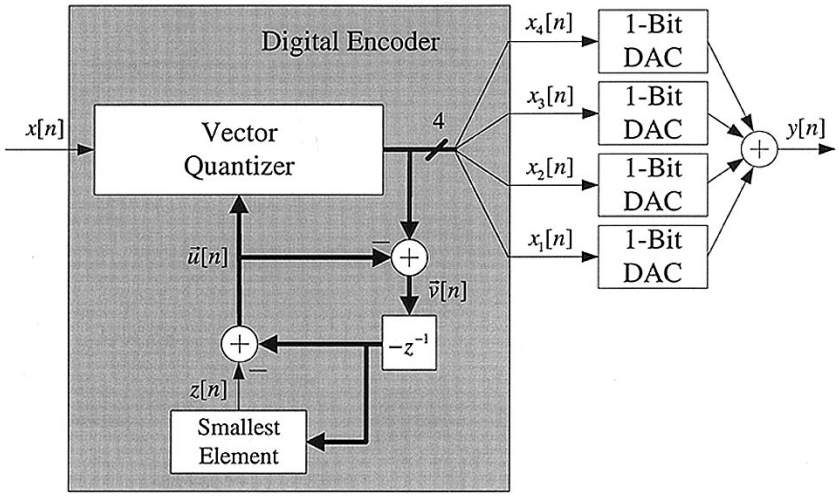


Fig. 2. The first-order, lowpass vector feedback DAC.

In *efficient* mismatch-shaping DACs (i.e., those for which it is not possible to achieve equivalent performance with fewer 1-bit DACs), none of the $N-1$ underlying sequences are constant for all DAC input sequences. To verify this assertion using Corollary 1, suppose $\phi_j[n] = K$ for all n, where K is some constant. Substituting this into (16) indicates

$$\sum_{k=1}^{N} a_{j,\,k} \cdot x_k[n] = K. \tag{18}$$

For A to be invertible, there must be some value k_0 such that $a_{j,\,k_0} \neq 0$. This and (18) imply that

$$x_{k_0}[n] = \sum_{\substack{k=1 \\ k \neq k_0}}^{N} d_k \cdot x_k[n] + \hat{K} \tag{19}$$

where $d_k \equiv -a_{j,\,k}/a_{j,\,k_0}$ and $\hat{K} \equiv K/a_{j,\,k_0}$. As a consequence of this linear dependence, an *equivalent* multibit DAC could be implemented using fewer than N 1-bit DACs; the only difference between the original and equivalent implementations would result from the 1-bit DAC errors in each. For example, if $x_{k_0}[n]$ were given by (19), then the k_0th 1-bit DAC could be removed by changing the nominal step sizes of the other 1-bit DACs according to the following: for $k \neq k_0$, $\Delta_k^{\text{new}} = \Delta_k^{\text{old}} + d_k \cdot \Delta_{k_0}^{\text{old}}$.

The following corollary is more convenient than the theorem or the first corollary for proving that the DAC noise from a given architecture *does not* have certain PSD properties.

Corollary 2: The multibit DAC in Fig. 1 produces DAC noise with a given set of PSD properties if and only if, for any selection of N constants, $f_1, \ldots, f_N$, there exist constants m and b, and a sequence $\varepsilon[n]$ with the given set of PSD properties such that

$$\sum_{i=1}^{N} f_i \cdot x_i[n] = mx[n] + b + \varepsilon[n]. \tag{20}$$

Proof: Sufficiency: Assume (20) holds. Let A be an $N \times N$ invertible matrix as described in Corollary 1. With $f_i \equiv \sum_{j=1}^{N-1} c_j \cdot a_{j,\,i}$ and $a \equiv m$, (16) and (20) imply (17). Thus, by Corollary 1, the multibit DAC produces DAC noise with the given set of PSD properties.

Necessity: Let $\phi_1[n], \ldots, \phi_{N-1}[n]$ be the $N-1$ sequences as described in the theorem and assume (6) and (7) hold. Substituting (6) into the left-hand side of (20) gives

$$\sum_{i=1}^{N} f_i \cdot x_i[n] = \sum_{j=1}^{N-1} \underbrace{\left(\sum_{i=1}^{N} f_i \cdot d_{i,\,j} \right)}_{\equiv c_j} \phi_j[n] + \underbrace{\left(\sum_{k=1}^{N} f_k \cdot m_k \right)}_{\equiv a_0} x[n]. \tag{21}$$

Upon substituting (7) into (21) and setting $m \equiv a + a_0$, (20) follows. ∎

Therefore, to show that the DAC noise does not have the given PSD properties, it is sufficient to find a linear combination of the digital encoder outputs that cannot be expressed as in (20). An example of this application is also shown in the following section.

IV. Architecture Analysis

The theorem and corollaries presented in the previous section are used in this section to analyze and compare several of the previously published multibit DAC architectures. Specifically, vector feedback, DWA, butterfly shuffler, tree structured, segmented butterfly shuffler, and partitioned DWA DAC architectures are considered.

A. Vector Feedback

A five-level (i.e., $N = 4$) example of the vector feedback DAC is shown in Fig. 2 [2]. Its input range is $\{-N/2, -N/2+1, \ldots, N/2\}$. Its 1-bit DACs all have the same nominal step size (i.e., $\Delta_i = \Delta_D$ for each i). The digital encoder consists of a *vector quantizer*, a *smallest-element* block, two vector adders, and a vector unit delay. The vector $\vec{u}[n]$ consists of N elements, the ith of which is associated with the ith output bit of the digital encoder. At each sample time n the vector quantizer determines

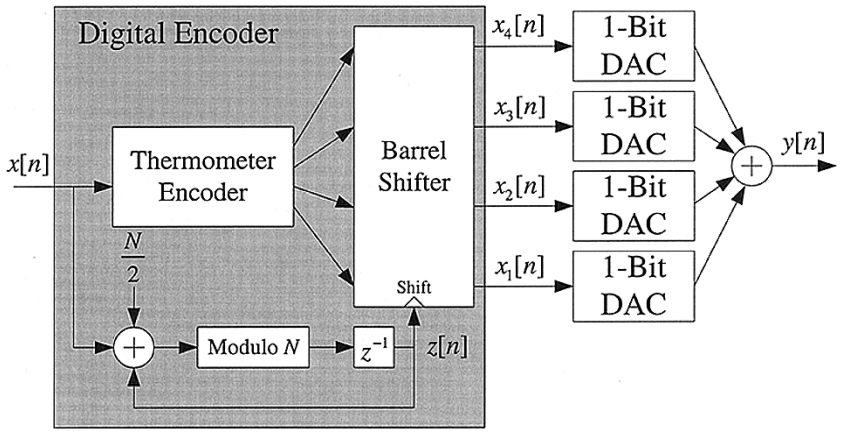


Fig. 3. The DWA DAC.

the $x[n] + N/2$ largest elements of $\vec{u}[n]$, and sets the associated output bits of the digital encoder high. It sets the remaining output bits low. The digital encoder calculates each element of $\vec{u}[n]$ as

$$u_i[n] = -v_i[n-1] - z[n] \tag{22}$$

where

$$v_i[n] = x_i[n] - u_i[n] \tag{23}$$

and $z[n] = \min_i\{-v_i[n-1]\}$, i.e., it is equal to the smallest element of $-\vec{v}[n-1]$.

To show that the feedback system within the digital encoder is stable, it is sufficient to show that $u_i[n]$ and $v_i[n]$ are bounded sequences for each value of i. Suppose that at some sample time, n_0, the largest element of $\vec{u}[n_0]$ has a value of $P \geq 1$. It follows from (22) that $u_i[n] \geq 0$ for each i and one element of $\vec{u}[n]$ equals zero for each n. The operation of the vector quantizer implies that $x_j[n_0] - x_i[n_0] = 1$ only when $u_j[n_0] \geq u_i[n_0]$. So (23) implies that

$$|v_i[n_0] - v_j[n_0]| \leq \max\{|u_i[n_0] - u_j[n_0]|, 1\} \leq P. \tag{24}$$

It follows from (22) that $u_j[n_0+1] - u_i[n_0+1] = v_i[n_0] - v_j[n_0]$, and since one element of $\vec{u}[n_0+1]$ is zero, (24) implies that $u_i[n_0+1] \leq P$ for each i. By induction, $u_i[n]$ must be a bounded sequence for each i, and, therefore, (23) implies that $v_i[n]$ must also be a bounded sequence for each i.

To apply Corollary 1, the invertible $N \times N$ matrix A must be chosen under the constraint that its Nth row is given by $a_{N,k} = 1$ for each k since $\Delta_i = \Delta_D$ for each i. For $j < N$, let

$$a_{j,k} = \begin{cases} 1, & \text{if } k = j+1 \\ -1, & \text{if } k = j \\ 0, & \text{otherwise} \end{cases} \tag{25}$$

which implies that

$$\phi_j[n] \equiv x_{j+1}[n] - x_j[n] \tag{26}$$

for $j = 1, \ldots, N-1$. To show that A is invertible as required by the corollary, it is sufficient to show that $A\vec{c} = \vec{0}$ only when $\vec{c} = \vec{0}$, where $\vec{c}$ is an N-length column vector and $\vec{0}$ is the vector consisting of only zeros. With the jth component of $\vec{c}$ denoted c_j, $A\vec{c} = 0$ implies that

$$c_{j+1} - c_j = 0 \tag{27}$$

for $j = 1, \ldots, N-1$, and

$$c_1 + \cdots + c_N = 0. \tag{28}$$

The difference equations characterized by (27) indicate that $c_i = c_j$ for each i and j. Upon substituting this into (28), it follows that $N \cdot c_j = 0$, which implies that $c_j = 0$ for each j. Therefore, A is an invertible matrix.

It is next shown that the choice of $\phi_j[n]$ given by (26) satisfies (17) with $a = 0$, $b = 0$, and an $\varepsilon[n]$ whose PSD is zero at $\omega = 0$. By virtue of Corollary 1, this implies that the PSD of the DAC noise also has a zero at $\omega = 0$, and, therefore, that the vector feedback DAC shown in Fig. 2 is a first-order mismatch-shaping DAC.

Substituting (22) into (23) gives $x_i[n] = v_i[n] - v_i[n-1] - z[n]$. With (26) this implies

$$\phi_j[n] = v_{j+1}[n] - v_{j+1}[n-1] - v_j[n] + v_j[n-1].$$

Therefore

$$\left|\sum_{m=0}^{n} \phi_j[m]\right| = |v_{j+1}[n] - v_j[n] - v_{j+1}[-1] + v_j[-1]|. \tag{29}$$

The partial sum in (29) is bounded for all n because $v_i[n]$ is a bounded sequence for each value of i. As shown in the Appendix, this implies that the PSD of $\phi_j[n]$ is zero at $\omega = 0$. It is also shown in the Appendix that any linear combination of such sequences has a PSD equal to zero at $\omega = 0$. Therefore, by Corollary 1, the DAC noise has this property too. A PSD plot of the DAC noise from behavioral simulations is provided in [2].

B. DWA

A five-level example of the DWA DAC is shown in Fig. 3 [3], [4]. Like the vector feedback DAC, its input range is $\{-N/2, -N/2+1, \ldots, N/2\}$, and all of its 1-bit DACs have the same nominal step size. The digital encoder consists of a thermometer encoder and a barrel shifter. Additionally, it consists of a modulo-N block, a unit delay, and an adder

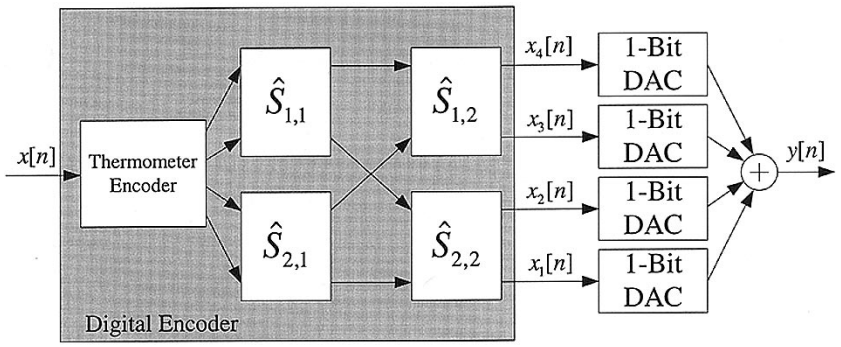


Fig. 4. The butterfly shuffler DAC.

that constitute a *modulo-N accumulator*. At each sample time n the thermometer encoder, whose N outputs are binary sequences, selects its bottom $x[n] + N/2$ outputs high and its remaining outputs low. The modulo-N accumulator output, $z[n]$, controls the operation of the barrel shifter as follows: with its inputs and outputs labeled 1 to N from bottom to top, the barrel shifter, at sample time n, routes input i to output $1+(z[n]+i-1) \bmod N$. Thus, the digital encoder outputs are generated by performing a modulo-N shift of the thermometer encoder outputs.

The values of $z[n]$ and $x[n]$ determine the digital encoder outputs at sample time n, and $z[n+1] = (x[n] + z[n] + N/2) \bmod N$. If $z[n] < z[n+1]$, then the high digital encoder outputs at time n are those numbered $z[n]+1, z[n]+2, \ldots, z[n+1]$, and the remaining outputs are low. However, if $z[n] > z[n+1]$, then the low digital encoder outputs at time n are those numbered $z[n+1]+1, z[n+1]+2, \ldots, z[n]$, and the remaining outputs are high. If $z[n+1] = z[n]$, then $x[n] = \pm N/2$, and all of the digital encoder outputs are either high or low at time n. Therefore, at each sample time, n, there is a contiguous segment of either high or low outputs of the digital encoder, and $z[n]$ and $z[n+1]$ determine the segment's starting and ending points.

To analyze the DAC noise using Corollary 1, let $\phi_j[n] \equiv x_{j+1}[n] - x_j[n]$ for $j = 1, \ldots, N-1$. As shown previously, these $N-1$ sequences are related to the digital encoder output sequences by an invertible matrix as required by the corollary.

As in the previous analysis, to show that the DAC noise PSD is zero at $\omega = 0$, it is sufficient to show that the partial sum of $\phi_j[n]$ is a bounded sequence. To show this, note that the $N-1$ sequences $\phi_1[n], \ldots, \phi_{N-1}[n]$ detect the edges—i.e., starting and ending points—of the contiguous segment of high or low digital encoder outputs. If $x[n] = \pm N/2$, there are no edges to detect and $\phi_j[n] = 0$ for each j. However, if $x[n] \neq \pm N/2$, $\phi_j[n]$ is nonzero only when j corresponds to an edge of the contiguous segment

$$\phi_j[n] = \begin{cases} -1, & \text{if } j = z[n] \\ 1, & \text{if } j = z[n+1] \\ 0, & \text{otherwise.} \end{cases} \tag{30}$$

This implies that the nonzero samples of $\phi_j[n]$ alternate between -1 and 1, and the partial sum of $\phi_j[n]$ is a bounded sequence

$$\left| \sum_{m=0}^{n} \phi_j[m] \right| \leq 1. \tag{31}$$

Therefore, the DAC noise PSD is also zero at $\omega = 0$. An example PSD plot of the DAC noise is provided in [4].

C. Butterfly Shuffler

An example of a five-level butterfly shuffler DAC is shown in Fig. 4 [5]. Like the previously analyzed DACs, its input range is $\{-N/2, -N/2+1, \ldots, N/2\}$, and all of its 1-bit DACs have the same nominal step size. Unlike the previously analyzed DACs, the butterfly shuffler DAC requires that N be a power of 2; i.e., $N = 2^b$, where b is a positive integer. The digital encoder consists of a thermometer encoder and N *swapper cells*, which are labeled $\hat{S}_{l,m}$ and positioned in a matrix with $l = 1, \ldots, 2^{b-1}$, and $m = 1, \ldots, b$, corresponding to the row and column numbers, respectively. The input and output sequences of each swapper cell are 1-bit sequences; the values of each are taken to be 1/2 and $-1/2$ at sample times when the sequence is high and low, respectively. At each sample time, n, each swapper cell determines its outputs by routing its inputs either straight through or swapped. The thermometer encoder, whose operation is described in the previous subsection, is not a necessary component as it can be replaced by any encoder that has N 1-bit outputs and ensures that exactly $x[n] + N/2$ of its outputs are high at each sample time, n.

Let $\hat{x}_{2l-1,m}[n]$ and $\hat{x}_{2l,m}[n]$ denote the top and bottom inputs of $\hat{S}_{l,m}$, respectively. Using $\hat{S}_{1,1}$ in Fig. 4 as an example

$$\hat{x}_{1,2}[n] = \tfrac{1}{2}\left(\hat{x}_{1,1}[n] + \hat{x}_{2,1}[n] + \hat{s}_{1,1}[n]\right) \tag{32}$$

and

$$\hat{x}_{3,2}[n] = \tfrac{1}{2}\left(\hat{x}_{1,1}[n] + \hat{x}_{2,1}[n] - \hat{s}_{1,1}[n]\right) \tag{33}$$

where $\hat{s}_{1,1}[n]$ is called a *swapper sequence*. This sequence is generated within $\hat{S}_{1,1}$ and is restricted to be 0 when $\hat{x}_{1,1}[n] = \hat{x}_{2,1}[n]$ and ± 1 otherwise. When $\hat{s}_{1,1}[n] = \pm 1$, the sign of the swapper sequence determines whether the swapper cell inputs are routed straight through or swapped. When $\hat{s}_{1,1}[n] = 0$, both swapper cell inputs are the same; therefore, both outputs are the same regardless of how the swapper cell routes its inputs. Thus, each swapper cell $\hat{S}_{l,m}$ uses its swapper sequence, $\hat{s}_{l,m}[n]$, as in (32) and (33) to determine its outputs. In the first-order butterfly shuffler DAC, each swapper cell alternates between swapping and not swapping so that

$$\left| \sum_{k=0}^{n} \hat{s}_{l,m}[k] \right| \leq 1 \tag{34}$$

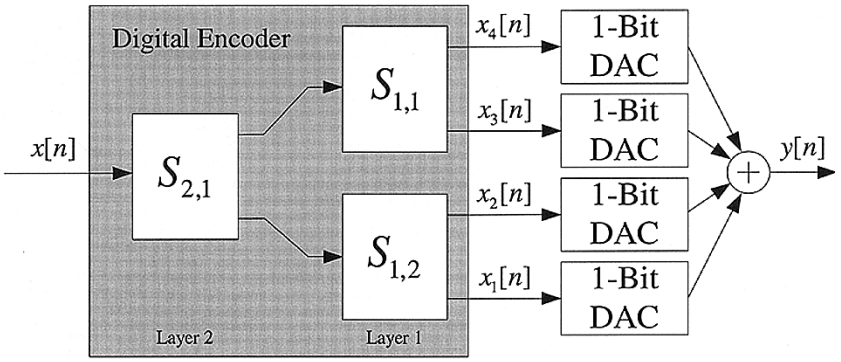


Fig. 5. The tree structured DAC.

which, as shown in the Appendix, implies that the PSD of each swapper sequence is zero at $\omega = 0$.

Generalizing (32) and (33) to the other swapper cells in Fig. 4, the top digital encoder output in the figure can be written as

$$x_4[n] = \tfrac{1}{4}\left(\sum_{k=1}^{4} \hat{x}_{k,1}[n] + \hat{s}_{1,1}[n] + \hat{s}_{2,1}[n]\right) + \tfrac{1}{2}\,\hat{s}_{1,2}[n]. \tag{35}$$

Since $x[n] + N/2$ of the thermometer encoder outputs are high at time n, it follows that $\sum_{k=1}^{4} \hat{x}_{k,1}[n] = x[n]$. This and (35) imply

$$x_4[n] = \tfrac{1}{4}\left(x[n] + \hat{s}_{1,1}[n] + \hat{s}_{2,1}[n]\right) + \tfrac{1}{2}\,\hat{s}_{1,2}[n]. \tag{36}$$

Therefore, the top digital encoder output is a linear combination of $x[n]$ and the swapper sequences. It follows by similar reasoning that this holds for every digital encoder output, and in general with $N = 2^b$

$$x_i[n] = x[n]/N + \sum_{l=1}^{2^{b-1}} \sum_{m=1}^{b} c_{l,m}^{(i)} \cdot \hat{s}_{l,m}[n] \tag{37}$$

where each $c_{l,m}^{(i)}$ is a constant that is either $\pm 1/2^{b-m+1}$ or 0.

To apply Corollary 1, once again let $\phi_j[n] \equiv x_{j+1}[n] - x_j[n]$ for $j = 1, \ldots, N-1$. As previously shown, the matrix that relates these $N-1$ sequences to the digital encoder outputs is invertible as required by the corollary. It follows from (34) and (37) that, for each j, $\phi_j[n]$ is a linear combination of swapper sequences whose PSDs are zero at $\omega = 0$, which, as shown in the Appendix, implies that the PSD of $\phi_j[n]$ is zero at $\omega = 0$. Therefore, the DAC noise PSD is also zero at $\omega = 0$. An example DAC noise PSD plot from this DAC is provided in [11].

D. Tree Structured

An example of a five-level tree structured DAC is shown in Fig. 5 [6]. Like the previously analyzed DACs, its input range is $\{-N/2, -N/2+1, \ldots, N/2\}$, and all of its 1-bit DACs have the same nominal step size. This DAC requires that N be a power of two: $N = 2^b$, where b is a positive integer. The digital encoder consists of *switching blocks*, which are labeled $S_{k,r}$, where $k = 1, \ldots, b$, denotes the layer number, and $r = 1, \ldots, 2^{b-k}$, denotes the depth in the layer. If the input to $S_{k,r}$ is denoted $x_{k,r}[n]$ and each sequence $x_i[n]$ is also denoted $x_{0,N+1-i}[n]$, the switching blocks are interconnected such that the top and bottom outputs of $S_{k,r}$ are $x_{k-1,2r-1}[n]$ and $x_{k-1,2r}[n]$, respectively. The outputs of $S_{k,r}$ are given by

$$x_{k-1,2r-1}[n] = \tfrac{1}{2}\left(x_{k,r}[n] + s_{k,r}[n]\right) \tag{38}$$

and

$$x_{k-1,2r}[n] = \tfrac{1}{2}\left(x_{k,r}[n] - s_{k,r}[n]\right) \tag{39}$$

where $s_{k,r}[n]$ is called the *switching sequence* and is generated within $S_{k,r}$.

Analogously to the butterfly shuffler DAC, the switching blocks in the first-order tree structured DAC ensure that

$$\left|\sum_{m=0}^{n} s_{k,r}[m]\right| \le 1 \tag{40}$$

which, as shown in the Appendix, implies that the PSD of $s_{k,r}[n]$ is zero at $\omega = 0$. By recursively solving the switching block outputs in (38) and (39) as functions of the switching sequences and the DAC input $x[n]$, it follows that

$$x_i[n] = x[n]/N + \sum_{k=1}^{b} \sum_{r=1}^{2^{b-k}} d_{k,r}^{(i)} \cdot s_{k,r}[n] \tag{41}$$

where each $d_{k,r}^{(i)}$ is a constant that is either $\pm 1/2^k$ or 0.

Once again, Corollary 1 can be applied by using the $N-1$ sequences $\phi_j[n] \equiv x_{j+1}[n] - x_j[n]$ for $j = 1, \ldots, N-1$. As previously shown, the $N-1$ sequences are generated by an invertible matrix as required by the corollary. The PSD of each $\phi_j[n]$ sequence is zero at $\omega = 0$ because, from (41), each sequence results from a linear combination of switching sequences whose PSDs are zero at $\omega = 0$. Therefore, the DAC noise PSD is also zero at $\omega = 0$, which is illustrated in the PSD plots provided in [6] and [12].

E. Qualitative Comparisons

Comparisons among mismatch-shaping DACs can be made using the necessary and sufficient condition presented in the theorem. One comparison can be made concerning how easily each of the four previously analyzed DACs combat harmonic distortion in its DAC noise. In the butterfly shuffler and tree structured DACs, the DAC noise is a linear combination of shaped sequences—i.e., swapper and switching sequences—that are generated within their digital encoders. Therefore, as shown in the Appendix, if these shaped sequences have bounded PSDs, then

their DAC noise PSDs are also bounded and thus do not contain spurious tones. This can be accomplished by incorporating randomness in the shaped sequences to prevent any tonal behavior. The relative ease for which this is accomplished is shown in [12] where pseudorandom sequences are employed by the switching blocks in both first- and second-order lowpass tree structured DACs to eliminate harmonic distortion in the DAC noise.

However, the vector feedback and DWA DACs obtain DAC noise with the given set of PSD properties without explicitly generating sequences with these properties. This indirect approach for spectrally shaping the DAC noise makes it more difficult to eliminate or reduce spurious tones. To remove spurious tones in the vector feedback DAC, randomness must somehow be incorporated into the vector quantizer's operation, but, to the knowledge of the authors, no such vector quantizer has been demonstrated to date. To remove or reduce spurious tones in the DWA DAC, its architecture must be changed. Most variants of the DWA DAC are designed to reduce, relative to the DWA DAC, the harmonic distortion in the DAC noise. Examples of such DWA variants are presented in [8], [13], and [14]. To successfully reduce harmonic distortion, each of these published first-order architectures requires that the multibit DAC input includes a random component—e.g., the quantization noise from a $\Delta\Sigma$ modulator. This is not required in the previously mentioned first-order, tree structured DAC whose DAC noise PSD is bounded regardless of the DAC input [15].

Another comparison can be made concerning the ease for which a mismatch-shaping DAC obtains *higher order*—i.e., greater than first order—spectral shaping of the DAC noise. Such DACs are desirable because the DAC noise in a higher order DAC usually has less signal-band power. This comparison does not include DWA because it is inherently a first-order DAC. The theorem states that, given the DAC input, $N-1$ sequences are required to generate the digital encoder outputs in a mismatch-shaping DAC. However, with $N=2^b$, where b is a positive integer, the butterfly shuffler DAC requires $b\cdot N/2$ swapper sequences, which, for $b>1$, are more than necessary as $b\cdot N/2 > N-1$. Additionally, as b increases, the number of extra sequences utilized by the DAC grows at a faster rate than an exponential function. Each swapper sequence depends on its swapper cell input, which depends on the DAC input. This dependence and the extra swapper sequences makes it difficult to ensure that each swapper sequence has the desired PSD properties in higher order implementations.

For example, to implement a second-order, lowpass butterfly shuffler DAC, it follows from [12] that each swapper sequence must satisfy the following:

$$\left|\sum_{j=0}^{n}\sum_{k=0}^{j}\hat{s}_{l,\,m}[k]\right| \le B \tag{42}$$

where B is a constant. Because the value of each swapper cell output is either $-1/2$ or $1/2$ at each sample time, n, it follows that

$$\hat{s}_{l,\,m}[n]=\begin{cases}\pm 1, & \text{if } \hat{x}_{2l-1,\,m}[n]\neq\hat{x}_{2l,\,m}[n]\\ 0, & \text{if } \hat{x}_{2l-1,\,m}[n]=\hat{x}_{2l,\,m}[n].\end{cases} \tag{43}$$

Therefore, if the N inputs to the column-one swapper cells are thermometer encoded as in Fig. 4, then the column-one swapper sequences are restricted as follows:

$$\hat{s}_{l,\,1}[n]=\begin{cases}\pm 1, & \text{if } x[n]=N/2-(2l-1)\\ 0, & \text{otherwise.}\end{cases} \tag{44}$$

At each sample time, at most one of the 2^{b-1} swapper sequences in the first column is nonzero; the choice of which is determined by the DAC input. As b increases, this dependence on the DAC input makes it more difficult for these swapper sequences to satisfy (42) and has prohibited the implementation of the second-order, lowpass butterfly shuffler DAC.

However, the vector feedback and tree structured DACs process N and $N-1$ internal sequences, respectively, to generate their digital encoder outputs. Because, for $b>2$, these DACs process fewer internal sequences than the butterfly shuffler DAC, their internal sequences and DAC noise have less dependence on the DAC input, which enables the implementation of higher order DACs. For example, in the tree structured DAC, the layer that directly processes the DAC input, layer b, only has one switching block as opposed to the 2^{b-1} swapper cells in the first column of the butterfly shuffler DAC. For the switching blocks presented in [12], the switching sequence is restricted as follows:

$$s_{k,\,r}[n]=\begin{cases}\pm 1, & \text{if } x_{k,\,r}[n]+2^{k-1} \text{ is odd}\\ 0, & \text{if } x_{k,\,r}[n]+2^{k-1} \text{ is even.}\end{cases} \tag{45}$$

Therefore, the switching sequence in layer b depends only on the parity of the DAC input, which is much less restrictive than the dependence exhibited by the column-one swapper sequences shown in (44). Examples of second-order lowpass implementations of the vector-feedback and tree structured DACs are presented in [16] and [17], respectively.

F. Segmented Butterfly Shuffler

The 65-level segmented butterfly shuffler DAC [7] shown in Fig. 6 uses 1-bit DACs with different nominal step sizes to reduce the complexity of the digital encoder relative to a nonsegmented 65-level butterfly shuffler DAC. The input to this DAC is in the range $\{-32,\ldots,32\}$. The digital encoder consists of 9- and 17-level butterfly shuffler digital encoders, a first-order digital $\Delta\Sigma$ modulator, a subtractor, and a gain element. The nominal step sizes of the 1-bit DACs that are driven by the 9- and 17-level butterfly shuffler digital encoders are Δ_D and $4\cdot\Delta_D$, respectively. The digital $\Delta\Sigma$ modulator quantizes with a step size of 4, and its output can be written as

$$u[n]=x[n]+\varepsilon[n]-\varepsilon[n-1] \tag{46}$$

where $\varepsilon[n]$ is the *quantization error* (i.e., the difference between the output and the input of the quantizer in the $\Delta\Sigma$ modulator). The DAC in Fig. 6 requires only 44 swapper cells compared to the 192 swapper cells required to implement a regular 65-level butterfly shuffler DAC.

To apply Corollary 1, $\phi_1[n],\ldots,\phi_{23}[n]$ must be derived using a 24 $\times$ 24 invertible matrix A whose 24th row must

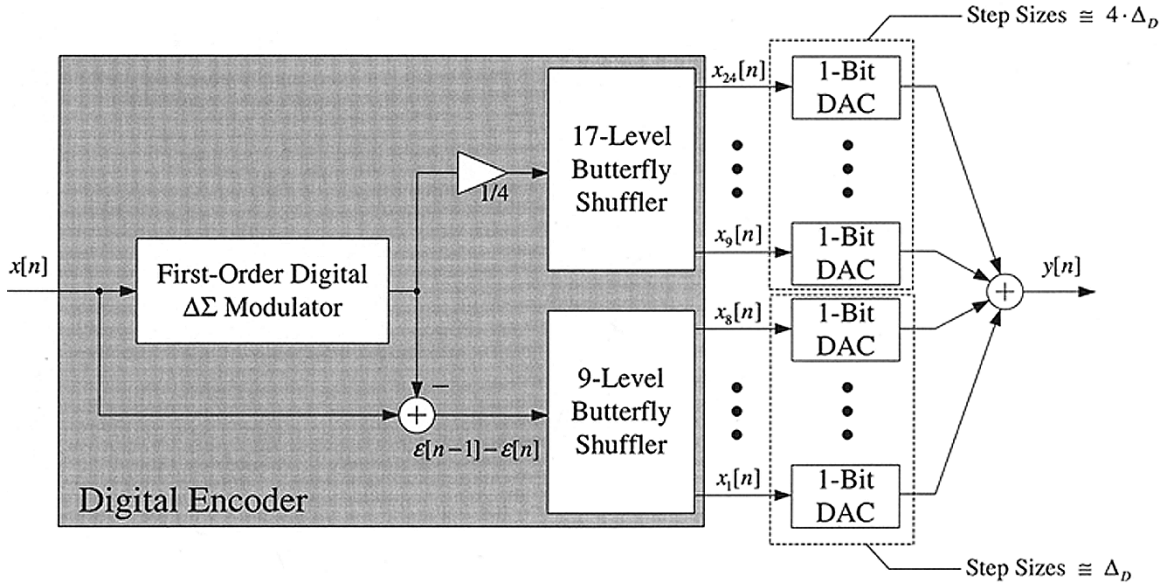


Fig. 6. The segmented butterfly shuffler DAC.

satisfy $a_{24,k} = \Delta_k/\Delta_D$. This implies that $a_{24,k} = 1$ for $k \leq 8$, and $a_{24,k} = 4$ otherwise. Let

$$a_{8,k} = \begin{cases} 1, & \text{if } k \leq 8 \\ 0, & \text{otherwise} \end{cases} \tag{47}$$

and for $k \neq 8, 24$, define $a_{j,k}$ as in (25). This implies that, for $j = 1, \ldots, 23$

$$\phi_j[n] = \begin{cases} \displaystyle\sum_{k=1}^{8} x_k[n], & \text{if } j = 8 \\ x_{j+1}[n] - x_j[n], & \text{otherwise.} \end{cases} \tag{48}$$

As previously described, A is invertible if $A\vec{c} = \vec{0}$ holds only when $\vec{c} = \vec{0}$. Given $A\vec{c} = \vec{0}$, it follows that

$$\sum_{j=1}^{8} c_j = 0 \tag{49}$$

$$\sum_{j=1}^{8} c_j + \sum_{l=9}^{24} 4 \cdot c_l = 0 \tag{50}$$

and, for $j \neq 8$

$$c_{j+1} - c_j = 0. \tag{51}$$

Substituting (49) into (50) yields

$$\sum_{j=9}^{24} c_j = 0. \tag{52}$$

From previous analysis, (49), (51), and (52) imply that $c_j = 0$ for each j, which proves that A is invertible as required by the corollary.

It is shown next that, for each j, the PSD of $\phi_j[n]$ is 0 at $\omega = 0$, which, by virtue of Corollary 1 and the results in the Appendix, implies that the DAC noise also has this property. For $j \neq 8$, it follows from (48) and the analysis of the butterfly shuffler DAC that $\phi_j[n]$ is a linear combination of swapper sequences whose PSDs are zero at $\omega = 0$. As shown in the Appendix, this implies that the PSD of $\phi_j[n]$ is also zero at $\omega = 0$.

Because the 1-bit DACs of a butterfly shuffler DAC have the same nominal step size, (2) implies that the sum of the outputs of a butterfly shuffler digital encoder equals its input. Therefore, $\phi_8[n]$, as given in (48), equals the input to the 9-level butterfly shuffler DAC, which, as shown in Fig. 6, is $\varepsilon[n-1] - \varepsilon[n]$. The partial sum of $\phi_8[n]$ then is a telescoping sum

$$\sum_{m=0}^{n} \phi_8[m] = \varepsilon[-1] - \varepsilon[n] \tag{53}$$

which is a bounded sequence because $\varepsilon[n]$ is the quantization error. As shown in the Appendix, this implies that the PSD of $\phi_8[n]$ is zero at $\omega = 0$. Therefore, for each j, the PSD of $\phi_j[n]$ is zero at $\omega = 0$, and the DAC noise also has this property.

G. Partitioned DWA

The partitioned DWA (P-DWA) DAC, shown in Fig. 7, was designed to not only suppress the DAC noise power near $\omega = 0$, but to reduce, in comparison to the DWA DAC, the spurious tones in the DAC noise . Its input range is $\{-16, -15, \ldots, 16\}$. All of its 1-bit DACs have the same nominal step size. The digital encoder consists of two 17-level DWA digital encoders and a *divide-by-two* block. The top output of the divide-by-two block is $x[n]/2$ rounded up to the nearest integer (i.e., $\lceil x[n]/2 \rceil$), and the bottom output is $x[n]/2$ rounded down to the nearest integer (i.e., $\lfloor x[n]/2 \rfloor$).

Corollary 2 is applied next to show that the DAC noise PSD is *not* zero at $\omega = 0$. Since the difference between the outputs of the divide-by-two block is one when $x[n]$ is odd and zero otherwise, it follows that

$$\sum_{i=17}^{32} x_i[n] - \sum_{j=1}^{16} x_j[n] = x[n] \bmod 2. \tag{54}$$

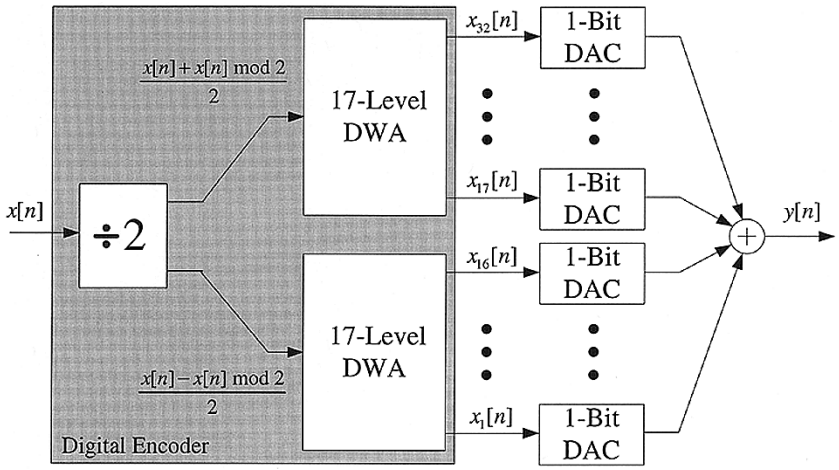


Fig. 7. The partitioned DWA DAC.

By Corollary 2, if the above linear combination *cannot* be written as $ax[n] + b + \varepsilon[n]$, where a and b are constants and the PSD of $\varepsilon[n]$ is zero at $\omega = 0$, then the DAC noise PSD is *not* zero at $\omega = 0$. Therefore, from (54), it is sufficient to show that, for some $x[n]$, the PSD of the sequence

$$\varepsilon[n] \equiv (x[n] \bmod 2) - (ax[n] + b) \tag{55}$$

is not zero at $\omega = 0$ for any choice of the constants a and b. Since

$$x[n] = 2\left\lfloor \frac{x[n]}{2} \right\rfloor + (x[n] \bmod 2). \tag{56}$$

Equation (55) can be written as

$$\varepsilon[n] = (1-a)(x[n] \bmod 2) - \left(2a\left\lfloor \frac{x[n]}{2} \right\rfloor + b\right). \tag{57}$$

The DAC input, $x[n]$, can be chosen so that $x[n] \bmod 2$ and $\lfloor x[n]/2 \rfloor$ in (57) are uncorrelated sequences whose PSDs are not zero at $\omega = 0$. For example, suppose $x[n]$ is a sequence of independent and identically distributed (i.i.d.) random variables that are uniformly distributed in the range $\{0, \ldots, 3\}$. In this case, $x[n] \bmod 2$ and $\lfloor x[n]/2 \rfloor$ are independent sequences that consist of i.i.d. random variables that are 0 or 1 with equal probability. For this $x[n]$, (57) implies that the PSD of $\varepsilon[n]$ is not zero at $\omega = 0$ for any selection of the constants a and b. Therefore, by Corollary 2, the DAC noise PSD is also not zero at $\omega = 0$.

Fig. 8 displays the output noise PSD from a behavioral simulation of a second-order, analog $\Delta\Sigma$ modulator that employs the P-DWA DAC. The $\Delta\Sigma$ modulator input was a -1 dB (relative to full scale) sinusoid with frequency $0.0015\, f_s$, where f_s is the sample rate. The PSD units are dB relative to Δ^2, where Δ is the step size of the analog-to-digital converter within the $\Delta\Sigma$ modulator. The frequency axis is normalized with respect to the sample rate. The 1-bit DAC errors were chosen as independent Gaussian random variables with a standard deviation of 1% of the 1-bit DACs nominal step size.

The output noise in the simulation includes the DAC noise and quantization noise. The simulation shows that, as a result of the DAC noise, the output noise PSD is not zero at $\omega = 0$. However, the simulation suggests that, compared to conventional DWA, the DAC noise in this implementation has less harmonic distortion. The reduced harmonic distortion is a result of the randomness in $x[n]$, which causes $(x[n] \bmod 2)/2$ to act as an additive and subtractive dither sequence that, as shown in Fig. 7, is fed into top and bottom DWA DACs, respectively.

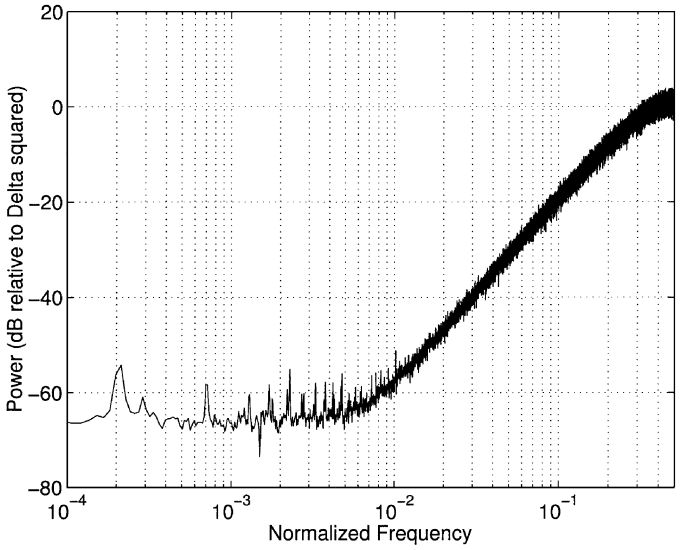


Fig. 8. The output noise PSD from a simulation of a second-order, analog $\Delta\Sigma$ modulator using the partitioned DWA DAC.

V. Conclusion

Necessary and sufficient conditions for mismatch shaping with a general multibit DAC have been presented, proved, and discussed. For the DAC noise to have certain PSD properties, the conditions show that there must be $N - 1$ underlying sequences in the general multibit DAC that, when linearly combined, produce a sequence that consists of an offset, a scaled version of the multibit DAC input, and another sequence that has the given PSD properties. As example applications, the conditions have been used to show that the DAC noise PSDs of five widely-used lowpass DACs are zero at $\omega = 0$ and that the DAC noise PSD of another lowpass DAC is not zero at $\omega = 0$. Additionally, the theory has been used to compare the ease for

which several DACs combat spurious tones in their DAC noise and obtain higher order shaped DAC noise.

APPENDIX

Two lemmas are presented below that supplement the analyses in Section IV. The first lemma proves that if a sequence $\gamma[n]$ has a partial sum that is a bounded sequence, then the PSD of $\gamma[n]$ is zero at $\omega = 0$. The second lemma proves an inequality for the PSDs that is used to show that an arbitrary linear combination of sequences whose PSDs are zero or bounded at a given frequency gives rise to a sequence whose PSD is also zero or bounded, respectively, at that frequency. It is assumed throughout that the PSDs exist for all sequences considered.

Lemma 1: Let $\gamma[n]$ be a sequence whose partial sum is bounded in magnitude by a constant B for all n; i.e.

$$\left|\sum_{m=0}^{n} \gamma[m]\right| \leq B \tag{58}$$

for all n. Then, the PSD of $\gamma[n]$ (if it exists) is zero at $\omega = 0$.

Proof: As proved in [18], the PSD of $\gamma[n]$ is given by

$$S_{\gamma,\gamma}\left(e^{j\omega}\right) = \lim_{M\to\infty} \frac{1}{M} E\left\{\left|\Gamma_M\left(e^{j\omega}\right)\right|^2\right\} \tag{59}$$

where $E\{\cdot\}$ is the expectation operator, and $\Gamma_M(e^{j\omega})$ is the M-point Fourier transform of $\gamma[n]$

$$\Gamma_M\left(e^{j\omega}\right) = \sum_{n=0}^{M-1} \gamma[n] e^{-j\omega n}. \tag{60}$$

Evaluating the PSD at $\omega = 0$ gives

$$S_{\gamma,\gamma}\left(e^{j0}\right) = \lim_{M\to\infty} \frac{1}{M} E\left\{\left|\sum_{n=0}^{M-1} \gamma_i[n]\right|^2\right\}. \tag{61}$$

However, from (58), the partial sum of $\gamma[n]$ in the above expression is bounded in magnitude by B; therefore

$$S_{\gamma,\gamma}\left(e^{j0}\right) \leq \lim_{M\to\infty} \frac{B^2}{M} = 0. \tag{62}$$

Because $S_{\gamma,\gamma}(e^{j\omega})$ is nonnegative for all ω, (62) implies that $S_{\gamma,\gamma}(e^{j0}) = 0$. ■

Lemma 2: If $S_{x,x}(e^{j\omega})$ and $S_{y,y}(e^{j\omega})$ are the PSDs of $x[n]$ and $y[n]$, respectively, and $z[n] = x[n] + y[n]$, then

$$S_{z,z}\left(e^{j\omega}\right) \leq 2\left(S_{x,x}\left(e^{j\omega}\right) + S_{y,y}\left(e^{j\omega}\right)\right) \tag{63}$$

where $S_{z,z}(e^{j\omega})$ is the PSD of $z[n]$.

Proof: Let $X_M(e^{j\omega})$, $Y_M(e^{j\omega})$, and $Z_M(e^{j\omega})$ be the M-point Fourier transforms of $x[n]$, $y[n]$, and $z[n]$, respectively—i.e.

$$X_M\left(e^{j\omega}\right) = \sum_{n=0}^{M-1} x[n] e^{-j\omega n} \tag{64}$$

and likewise for the Fourier transforms of $y[n]$ and $z[n]$. The Cauchy–Schwartz inequality implies that

$$|a+b|^2 \leq 2\left(|a|^2 + |b|^2\right) \tag{65}$$

where a and b are complex numbers. Therefore, it follows from the linearity of the Fourier Transform that, for every ω

$$\left|Z_M\left(e^{j\omega}\right)\right|^2 \leq 2\left(\left|X_M\left(e^{j\omega}\right)\right|^2 + \left|Y_M\left(e^{j\omega}\right)\right|^2\right). \tag{66}$$

As shown in [18]

$$S_{z,z}\left(e^{j\omega}\right) = \lim_{M\to\infty} \frac{1}{M} E\left\{\left|Z_M\left(e^{j\omega}\right)\right|^2\right\} \tag{67}$$

and likewise for PSDs of $x[n]$ and $y[n]$, where $E\{\cdot\}$ is the expectation operator. Therefore, (66), (67), and the linearity of the expectation operator imply (63). ■

Therefore, it follows from (63) that if, at some frequency ω_o, $S_{x,x}(e^{j\omega_o}) = 0$, and $S_{y,y}(e^{j\omega_o}) = 0$, then $S_{z,z}(e^{j\omega_o}) = 0$ because the PSD is always nonnegative. Thus, the sum of two sequences whose PSDs are zero at some frequency gives rise to a sequence whose PSD is also zero at that frequency. Additionally, if the PSDs of $x[n]$ and $y[n]$ are bounded functions—i.e., there exists a constant B such that $S_{x,x}(e^{j\omega}) \leq B$, and $S_{y,y}(e^{j\omega}) \leq B$ for all ω—then (63) implies that the PSD of $z[n]$ is also a bounded function: $S_{z,z}(e^{j\omega}) \leq 4B$. Therefore, by mathematical induction, any linear combination of sequences whose PSDs are zero or bounded at a given frequency give rise to another sequence whose PSD is also zero or bounded, respectively, at that frequency.

REFERENCES

[1] L. Hernández, "A model of mismatch-shaping D/A conversion for linearized DAC architectures," *IEEE Trans. Circuits Syst. I*, vol. 45, pp. 1068–1076, Oct. 1998.

[2] R. Schreier and B. Zhang, "Noise-shaped multibit D/A converter employing unit elements," *Electron. Lett.*, vol. 31, no. 20, pp. 1712–1713, Sept. 28, 1995.

[3] 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.

[4] R. T. Baird and T. S. Fiez, "Linearity enhancement of multibit ΔΣ A/D and D/A converters using data weighted averaging," *IEEE Trans. Circuits Syst. II*, vol. 42, pp. 753–762, Dec. 1995.

[5] R. W. Adams and T. W. Kwan, "Data-directed scrambler for multibit noise shaping D/A converters," U.S. Patent no. 5 404 142, Apr. 4, 1995.

[6] I. Galton, "Spectral shaping of circuit errors in digital-to-analog converters," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 808–817, Oct. 1997.

[7] R. Adams, K. Nguyen, and K. Sweetland, "A 113-dB SNR oversampling DAC with segmented noise-shaped scrambling," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1871–1878, Dec. 1998.

[8] K. Vleugels, S. Rabii, and B. A. Wooley, "A 2.5 V sigma–delta modulator for broadband communications applications," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1887–1899, Dec. 2001.

[9] L. Hernández, "Binary weighted D/A converters with mismatch-shaping," *Electron. Lett.*, vol. 33, no. 24, pp. 2006–2008, Nov. 20, 1997.

[10] J. Steensgaard, U. Moon, and G. C. Temes, "Mismatch-shaping switching for two-capacitor DAC," *Electron. Lett.*, vol. 34, no. 17, pp. 1633–1634, Aug. 20, 1998.

[11] R. Schreier, "Mismatch-shaping digital-to-analog conversion," in *Proc. 103rd Convention Audio Engineering Soc.*, Preprint, Sept. 26–29, 1997.

[12] J. Welz, I. Galton, and E. Fogleman, "Simplified logic for first-order and second-order mismatch-shaping digital-to-analog converters," *IEEE Trans. Circuits Syst. II*, vol. 48, Nov. 2001.

[13] I. Fujimori, L. Longo, A. Hairapetian, K. Seiyama, S. Kosic, J. Cao, and S. Chan, "A 90 dB SNR, 2.5 MHz output-rate ADC using cascaded multibit Delta–Sigma modulation at 8× oversampling ratio," *IEEE J. Solid-State Circuits*, vol. 35, pp. 1820–1828, Dec. 2000.

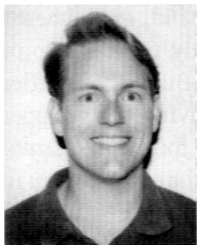
[14] R. Radke, A. Eshraghi, and T. Fiez, "A spurious-free Delta–Sigma DAC using rotated data weighted averaging," in *Proc. IEEE Custom Integrated Circuits Conf.*, May 1999, pp. 125–128.

[15] J. Welz and I. Galton, "The mismatch-noise PSD from a tree-structured DAC in a second-order Δ–Σ modulator with a midscale input," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 4, May 7–11, 2001.

[16] A. Yasuda, H. Tanimoto, and T. Iida, "A third-order $\Delta\Sigma$ modulator using second-order noise-shaping dynamic element matching," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1879–1886, Dec. 1998.

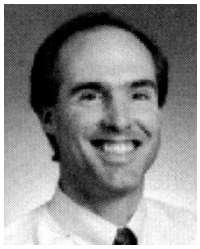
[17] E. Fogleman, J. Welz, and I. Galton, "An audio ADC Delta–Sigma modulator with 100 dB SINAD and 102 dB DR using a second-order mismatch-shaping DAC," *IEEE J. Solid State Circuits*, vol. 36, pp. 339–348, Mar. 2001.

[18] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1996.



Jared Welz (S'98–M'02) received the B.S.E.E. degree from the University of California, Irvine, in 1993, the M.S.E.E. degree from the University of California, Los Angeles, in 1994 with an emphasis in communication theory, and the Ph.D. degree in electrical engineering from the University of California, San Diego, La Jolla, in 2002.

From 1994 and 1997, he was with AirTouch International (now Vodafone AirTouch), Pacific Bell Wireless (now Cingular Wireless), and L.A. Cellular (now AT&T Wireless). From 1997 to 2002, he was a Graduate Student Researcher at the University of California, San Diego. Currently, he is with the Custom IC Design Group, Northrop Grumman Space Technology, Redondo Beach, CA. His research interests include data converters, signal processing, communication systems, and probability theory.



Ian Galton (M'92) received the Sc.B. degree from Brown University, Providence, RI, in 1984, and the M.S. and Ph.D. degrees from the California Institute of Technology, Pasadena, in 1989 and 1992, respectively, all in electrical engineering.

Previously, he was with the University of California, Irvine, the NASA Jet Propulsion Laboratory, Pasadena, CA, Acuson Corporation, Mountain View, CA and Mead Data Central. Since 1996, he has been an Associate Professor of electrical engineering at the University of California, San Diego, La Jolla, where he teaches and conducts research in the field of mixed-signal integrated circuits and systems for communications. His research involves the invention, analysis, and integrated circuit implementation of key communication system blocks such as data converters, frequency synthesizers, and clock recovery systems. The emphasis of the research is on the development of digital signal processing techniques to mitigate the effects of nonideal analog circuit behavior with the objective of generating enabling technology for highly integrated, low-cost communication systems. He regularly acts as Consultant to several communications and semiconductor companies and teaches portions of various industry-oriented short courses on the design of data converters, PLLs, and wireless transceivers, and has served as a Director on numerous Corporate Boards and several Technical Advisory Boards.

Dr. Galton is a member of the IEEE Circuits and Systems Society Board of Governors and is the Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND DIGITAL SIGNAL PROCESSING.