LINEARITY ENHANCEMENT OF MULTIBIT ΔΣ MODULATORS USING PSEUDO DATA-WEIGHTED AVERAGING

Anas A. Hamoui and Ken Martin

University of Toronto, Department of Electrical and Computer Engineering
10 King's College Road, Toronto, Ontario, Canada, M5S 3G4
E-mail:{hamoui,martin}@eecg.toronto.edu

ABSTRACT

The occurrence of inband signal-dependent tones, when using data-weighted averaging (DWA) in multibit ΔΣ modulators, degrades the spurious-free dynamic range and can preclude using DWA at low oversampling ratios (OSR). This paper proposes a simple technique, called Pseudo DWA, to solve the tone problem in DWA. Pseudo DWA is implemented in a 3rd-order ΔΣ A/D converter with a 32-level internal D/A converter (DAC), a 16x OSR, and a 2-MHz conversion bandwidth. Simulations show that, with a 0.5% DAC element mismatch, Pseudo DWA attenuates the DWA tones at the cost of only 1-dB drop in the signal-to-noise ratio. Existing techniques are also compared using simulations.

I. INTRODUCTION

Oversampled ΔΣ A/D converters (ADC) are well known for their ability to achieve a high-resolution A/D conversion of medium-to-low frequency signals. Extending these architectures to wideband applications requires lowering the oversampling ratio (OSR) in order to be realizable in the available IC technology and to meet the power budget. In a single-loop ΔΣ modulator (Fig. 1), the loss in dynamic range (DR) due to the lowering of the OSR can be compensated for by increasing the order of noise-shaping L and/or the resolution of the internal quantizer N [1]. The impact of increasing L on the DR diminishes significantly as the OSR is reduced. In contrast, the effectiveness of increasing N is independent of the OSR. Other advantages of multibit quantization include enhanced modulator stability and reduced loop-filter requirements [2,3]. However, because errors due to nonidealities in the feedback D/A converter (DAC) add directly to the input signal (and, therefore, are not shaped by the ΔΣ modulator), the linearity of a multibit ΔΣ modulator is limited by the linearity of its multibit internal DAC. High DAC linearity requires precise matching of the DAC unit elements (typically capacitors or current sources). Rather than using special fabrication processes or laser trimmed components to improve the element matching, two signal-processing strategies have been developed to correct for the DAC nonlinearity due to static element mismatch errors [1]:

1) Dynamic Element Matching (DEM): An algorithm, implemented using the element selection logic (Fig. 1), selects different DAC unit elements to represent a given digital code at different times, thereby translating the element mismatch from a dc error into a wideband high-pass-shaped noise. This shaped mismatch noise can then be filtered out by the decimation filter. DEM techniques differ in their effectiveness in shaping the spectrum of the DAC mismatch errors and in the complexity of the element selection logic. Zero-order [4], 1st-order [5-7], and 2nd-order [8-10] mismatch-shaping algorithms have been developed.

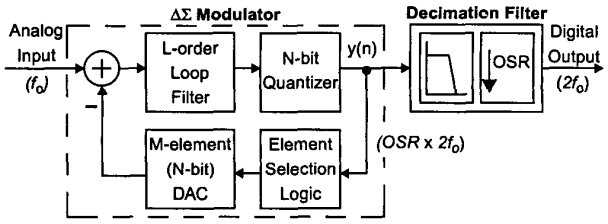


Fig. 1: Block diagram of a single-loop oversampled ΔΣ ADC using DEM in the feedback DAC.

More detailed description of DEM schemes is given in [2] and [10].

2) Calibration/Correction: Compared to off-line calibration techniques [11-12], background (on-line) calibration schemes [13-17] have the advantage of being robust to environmental changes. These schemes use analog calibration [13-14] or digital correction [15-17] to improve the effective DAC linearity. A combination of digital correction and DEM is used to acquire and correct for the DAC errors in [17].

Data-weighted averaging (DWA) [6] is a highly practical DEM technique, especially when the number of DAC elements is large. Compared to DEM techniques such as DWA, background calibration schemes are more expensive to implement in terms of system design complexity, hardware requirement, and power consumption. However, these calibration schemes are more suitable for achieving high resolution (≥ 15 bits) in high-speed (sampling frequency ≥ 100 MHz) ΔΣ ADCs operating at low OSR (≤ 8) because: 1) the effectiveness of DEM techniques falls drastically at low OSR; and 2) the delay introduced by the element selection logic in the feedback loop of the ΔΣ modulator can limit the maximum achievable clock speed.

The focus of this paper is on DWA. In section II, the tone behavior of DWA is briefly discussed. A technique to reduce the signal-dependent tones in DWA is proposed in Section III. The performance of the proposed technique is discussed in Section IV and compared to previously reported schemes. Its implementation in a test-chip ΔΣ ADC is described in Section V.

II. TONE BEHAVIOR OF DWA

DWA achieves first-order shaping of the DAC mismatch errors [2,8,18], but can introduce inband signal-dependent tones in the modulator's output spectrum (as depicted in Fig. 2a) [6,19].

In DWA, the unit elements participating in the D/A conversion are sequentially selected from the DAC array, beginning with the next available unused element. Because the same set of DAC elements is used cyclically and repeatedly under the guide of a single pointer (hereafter called the index pointer) [19], the element

mismatch errors translate to tones at the DAC output when the DAC input codes have a periodic pattern. These tones add directly to the modulator's input signal and appear unshaped at the modulator's output. Out-of-band tones generated by DWA may then fold back to the signal band due to modulation by the modulator's output waveform. The tone behavior of DWA depends on the number of DAC elements, the pattern of the DAC element mismatch, the amplitude and frequency of the modulator's input signal, and various circuit parameters which affect the modulator's output waveform [19-20].

For example, consider the case when the consecutive input codes to the M-element DAC in Fig. 1 have the same value Y (i.e. $y(n) = Y$). As is common, define the mismatch error ε_i in the DAC unit element U_i (i= 0, 1, ... , $M-1$) as

$$U_i = U_{mean}\ (1+\varepsilon_i), \text{ where } U_{mean} = \frac{1}{M}\sum_{i=0}^{M-1} U_i. \quad (1)$$

The DWA algorithm will select DAC unit elements in a rotational manner, making a complete rotation (with the index pointer returning to its exact starting point) every M/r clock cycles, where r is the greatest common divisor of Y and M. As a result, the DAC mismatch noise will be a periodic sequence

$$\left\{ \sum_{i=0}^{Y-1} \varepsilon_i\ ,\ \sum_{i=Y}^{2Y-1} \varepsilon_i\ ,\ \ldots\ ,\ \sum_{i=M-Y}^{M-1} \varepsilon_i\ ,\ \sum_{i=0}^{Y-1} \varepsilon_i\ ,\ \ldots \right\} \quad (2)$$

of period $(M/r)\ T_s$, and its power spectrum will take the form of tones at frequencies [20]

$$f_{tone} = k\,\frac{r}{M}\,f_s\ ,\quad k= 1, 2, \ldots \quad (3)$$

where f_s is the modulator's sampling frequency and $T_s = 1/f_s$. Thus, in general, for an M-element DAC, the lowest tone frequency can be at f_s/M with DWA. If the maximum possible input frequency to the modulator is $f_B = f_s/(2\ OSR)$, then the constraint that the k-th harmonic of the modulator's input signal not fall back into the signal bandwidth after being modulated by the tone at f_s/M requires that [19]

$$\frac{f_s}{M} - k\,f_B > f_B \ \Leftrightarrow\ OSR > \left(\frac{1+k}{2}\right)M\,. \quad (4)$$

In summary, the tone problem of DWA degrades the spurious-free dynamic range (SFDR) of the ΔΣ modulator and can preclude using DWA at low OSR. In the following, a simple technique will be proposed to solve this problem.

III. PSEUDO DWA

In the implementation of DWA, the index pointer ptr (i.e. the address of the next available unused element) is stored in a digital register. For an M-element DAC, the DAC elements selected at time n are those from $ptr(n)$ to $(ptr(n)+y(n)-1) \bmod M$, by increasing order. Every clock cycle, the index pointer is incremented modulus M by the DAC input code $y(n)$:

$$ptr(n+1) = (ptr(n)+y(n)) \bmod M,\ 0 \le ptr \le (M-1). \quad (5)$$

The technique proposed in this paper, called **Pseudo DWA**, modifies the DWA scheme by periodically inverting the least significant bit (LSB) of the DAC input code $y(n)$ used to update the index pointer in equation (5). Let $\mathbf{n_{inv}}$ denote the number of clock cycles between each LSB inversion of the DAC input code used to update the index pointer. The element selection process in Pseudo DWA is essentially similar to conventional DWA except that, every n_{inv} clock cycles, a DAC element is either re-selected or skipped depending on whether the previous DAC input code was odd or even. For example, assume that $ptr(n+1)$ has a value K in equation (5). If the corresponding $y(n)$ is even, its LSB inversion will increment $ptr(n+1)$ by 1. As a result, on the next clock cycle (i.e. at time $n+1$), the Pseudo-DWA algorithm will select DAC elements starting with element $K+1$ (i.e. element K is skipped). Alternatively, if $y(n)$ is odd, its LSB inversion will decrement $ptr(n+1)$ by 1. As a result, on the next clock cycle, the Pseudo-DWA algorithm will select DAC elements starting with element $K-1$ (i.e. element $K-1$ is re-selected).

This simple modification to DWA breaks the cyclic nature of the element selection process and, hence, reduces the tone behavior.

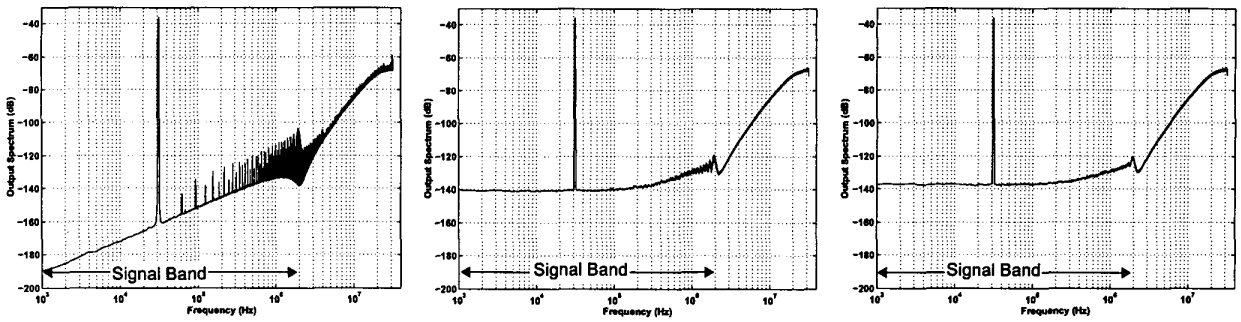
IV. PERFORMANCE OF PSEUDO DWA

The Pseudo-DWA algorithm is implemented in a 3rd-order ΔΣ ADC with a 5-bit quantizer, a 31-element internal DAC, a 16x OSR, a 2-MHz conversion bandwidth (f_B), and a 64-MHz sampling frequency (f_s). The ΔΣ ADC is based on a discrete-time single-loop ΔΣ modulator with a feed-forward architecture and with the quantization-noise transfer function having one zero at dc and two complex-conjugate zeros at the signal-band edge f_B [3]. The results reported in this paper correspond to the average of 100 SIMULINK simulations assuming a random DAC element mismatch of 0.5% (1σ). These simulations account for the finite dc gain of the integrators' opamps (150 V/V), and include the quantization noise and the DAC mismatch noise, but no other analog device noise. Since simulations with an ideal internal DAC showed no obvious inband tones in the modulator's output spectrum, it can be assumed that, with a non-ideal DAC, any inband tones are generated by the DAC element mismatch errors.

The choice of n_{inv} is a compromise between linearity and resolution. If n_{inv} is too large, the signal-dependent tones will not be eliminated: $n_{inv} = \infty$ corresponds to conventional DWA. If n_{inv} is too small, different DAC elements will be used at significantly different rates. Simulations show that this increases the inband mismatch noise and degrades the signal-to-noise-plus-distortion ratio (SNDR). For the ΔΣ modulator described above, simulations indicate that an n_{inv} between 64 and 128 is a good choice for preventing signal-dependent tones without sacrificing the SNDR. Fig. 2a shows the output spectrum of the ΔΣ modulator with DWA where strong inband tones are present. With Pseudo DWA and n_{inv} = 128 (Fig. 2b) or n_{inv} = 64 (Fig. 2c), these tones are smoothed in all frequencies and no notable tones are present in the signal band. This is accompanied with some increase in the inband noise. However, as depicted in Fig. 3, the degradation in SNDR compared to DWA is within 1 dB for n_{inv} = 128 (1.8 dB for n_{inv} = 64) over the full range of input signal amplitudes. This implies that, if the DAC mismatch noise floor is below the overall noise floor of the ADC, it is beneficial to use Pseudo DWA and improve the SFDR.

A number of other techniques have been proposed to solve the tone problem in DWA. These techniques will be first categorized before comparing their performance with Pseudo DWA:

1) Dithering [6]: Adding random dither can randomize the DAC input codes and spread the power of the DAC mismatch tones over all frequencies. However, dithering degrades the SNDR, reduces the DR, and can destabilize the ΔΣ modulator.



(a) DWA (b) Pseudo DWA with $n_{inv} = 128$ (c) Pseudo DWA with $n_{inv} = 64$

Fig. 2: Output spectrum of the ΔΣ modulator using: a) DWA; b) Pseudo DWA- $n_{inv} = 128$; and c) Pseudo DWA- $n_{inv}=64$. (V_{in} = - 30 dB, $f_{in} = f_s/2048$)

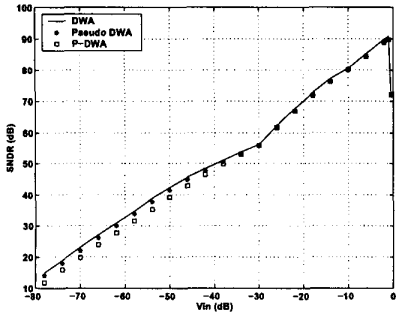


Fig. 3: SNDR vs. input signal amplitude. ($f_{in} = f_s / 2048$)

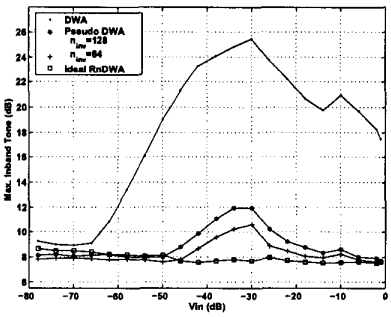


Fig. 4: Maximum inband tone vs. input signal amplitude. ($f_{in} = f_s / 2048$)

DAC Linearization	Peak SNDR (dB)
Ideal DAC	92.8
No DWA	60.4
DWA	90.9
Pseudo DWA, $n_{inv} = 128$	89.9
Pseudo DWA, $n_{inv} = 64$	89.1
P-DWA [23]	89.7
Bi-DWA [22]	85.8
Ideal RnDWA [21]	83.2
IDWA [20]	82.0

Table 1: Peak SNDR. ($f_{in} = f_s / 2048$)

2) Incremental DWA (IDWA) [20]: By using an extra DAC unit element in the DWA process, the DAC mismatch tones can be shifted away from $f_s/2$, thereby reducing the power of the aliased tones in baseband. However, for an M-element DAC, an $OSR > (M+1)$ is required to ensure that the shifted DAC tones fall outside the signal band.

3) Using multiple data-directed index pointers:

a) *Bi-Directional DWA (Bi-DWA)* [22]: Bi-DWA switches the direction of rotation in the DWA algorithm every clock cycle, using separate index pointers for the odd and even cycles.

b) *Partitioned DWA (P-DWA)* [23]: An M-element DAC is partitioned into two $M/2$-element sub-DACs. Its input digital code is also divided by 2, with the quotient plus remainder applied to the first sub-DAC and only the quotient applied to the second sub-DAC. DWA is then performed independently on each sub-DAC. Hence, when the input code is odd, one more element is selected in the first sub-DAC than in the second sub-DAC.

4) Randomizing the index pointer:

a) *Rotated DWA (RDWA)* [24]: Different circular patterns for the selection of DAC unit elements are kept in memory. DWA is performed by following one circular pattern for a limited time, and then switching randomly to a different pattern. Although RDWA is effective in reducing the tones present in DWA, its hardware becomes prohibitively large for DACs with 16 elements or more.

b) *Randomized DWA (RnDWA)* [21]: RnDWA selects DAC unit elements as in DWA except that, after each complete rotation (i.e. when the index pointer returns to its exact starting point), the index pointer is randomly shifted so that the next complete rotation begins from a new starting point. In the following, an ideal implementation (**ideal RnDWA**), where the new starting point of the index pointer is selected from a uniformly-distributed random sequence, is assumed. The effect of using a finite-length pseudo-random sequence for that purpose in a practical implementation should be verified in simulations.

c) *Pseudo DWA* (described in this paper).

Table 1 compares the peak SNDR for the various techniques described above. In DWA, the efficiency of the mismatch errors averaging to zero is determined by the rate of using each DAC element [6]. The rotation within M elements instead of $M+1$ in IDWA, the switching between separate pointers in Bi-DWA, and the frequent jumps in the index pointer in RnDWA, reduce this rate and degrade the SNDR. For example, let k be the number of complete rotations before shifting the index pointer in RnDWA. Then, having $k > 1$ (rather than $k = 1$ as in [21]) improves the efficiency of mismatch shaping (and, hence, the SNDR) but can degrade the tone behavior. For P-DWA, Fig. 3 shows that the degradation in SNDR compared to DWA is within 1.2 dB for input signal levels above -40dB, but is over 3 dB for lower levels.

Fig. 4 shows the maximum inband tone relative to the average tone power. Compared to DWA, reductions of about 14 dB, 15 dB, and 18 dB in the maximum inband tone are achieved using Pseudo DWA with n_{inv} = 128, Pseudo DWA with n_{inv} = 64, and ideal RnDWA, respectively. It is important to point out that neither Pseudo DWA nor the techniques in [20-23], except for ideal RnDWA, will completely eliminate the DAC inband tones (i.e. achieve a very smooth DAC noise floor) for all input signal levels. However, assuming a 0.5% DAC element mismatch in the ΔΣ

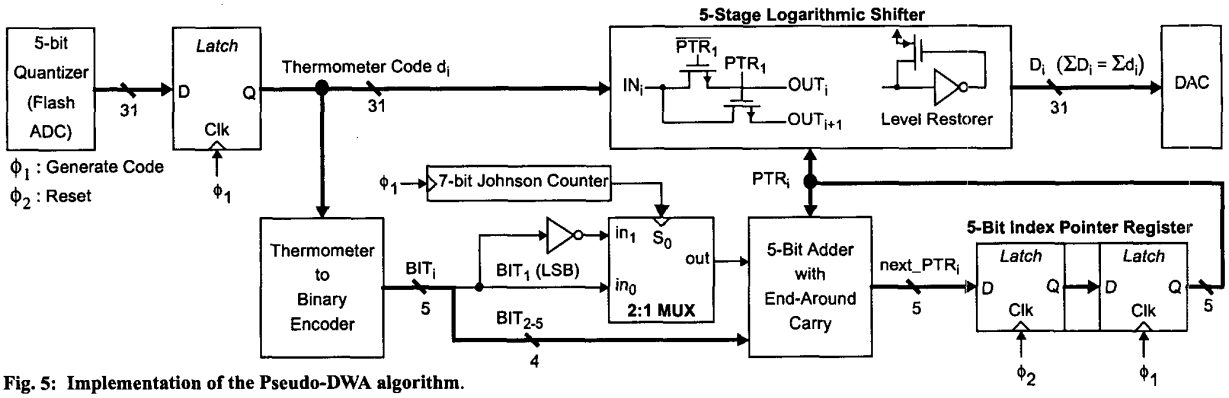


Fig. 5: Implementation of the Pseudo-DWA algorithm.

ADC described above, simulations show that the largest inband tone with Pseudo DWA is well below the level of the kT/C noise floor for the ADC. Thus, these inband tones will not be visible in the ADC output spectrum and will not degrade its performance.

V. IMPLEMENTATION OF PSEUDO DWA

The implementation of any DEM algorithm is critical for the achievable clock speed because the DEM block (element selection logic in Fig. 1) adds additional delay in the feedback loop of the $\Delta\Sigma$ modulator. Typically, in a switched-capacitor implementation of a $\Delta\Sigma$ ADC, the N-bit quantizer is realized using a flash ADC and a 2-phase non-overlapping clock is used. Only half a clock period (phase ϕ_1) is available for the output thermometer-code of the flash ADC to be generated, processed by the DEM block, and D/A converted by the DAC [25]. Therefore, it is very important to minimize the delay introduced by the DEM block.

The implementation of the Pseudo-DWA algorithm in the $\Delta\Sigma$ ADC described above is shown in Fig. 5. A 5-stage logarithmic shifter is used to provide the rotation of the 31-digit thermometer code as required by the Pseudo-DWA algorithm. The 5-bit control signal of the shifter corresponds to the index pointer and should be stable during phase ϕ_1, when the thermometer code ripples through the shifter. The shifter is implemented using only NMOS transistors, with level restorers after the 3-rd and 5-th stages to achieve a minimal delay. The index pointer is updated every clock cycle as follows: 1) an encoder converts the 31-digit thermometer code to a 5-bit binary code; 2) a 5-bit adder, with end-around carry, increments the index pointer modulus 31 by the quantizer output code and generates the next index pointer (i.e. the index pointer to be used at the next ϕ_1); 3) Every 128 clock cycles, the LSB of the quantizer output code is inverted before updating the index pointer register. A 7-bit Johnson counter and a 2:1 mux are used to control the timing of the LSB inversion. This is the only additional hardware required to implement Pseudo DWA instead of DWA.

VI. CONCLUSION

Pseudo DWA was proposed to remedy the tone behavior of DWA without sacrificing the SNDR. It requires only minimal additional digital signal processing. Simulations confirm the improved accuracy achievable by the proposed technique at low OSR.

VII. REFERENCES

[1] S. R. Norsworthy, R. Schreier, and G. C. Temes, Eds., *Delta-Sigma Data Converters Theory, Design, and Simulation*. New York: IEEE Press, 1996.

[2] R. Schreier, "Mismatch-shaping digital-to-analog conversion," in *103rd Convention Audio Engineering Society*, preprint no. 4529, Sept. 1997.

[3] A. A. Hamoui and K. Martin, "A higher-order multibit $\Delta\Sigma$ modulator for wideband low-distortion A/D converters," submitted to *IEE Electronics Lett.*

[4] L. R. Carely, "A noise-shaping coder topology for 15+ bit converters," *IEEE J. Solid-State Circuits*, vol. 24, no. 2, pp. 267-273, Apr. 1989.

[5] B. H. Leung and S. Sutarja, "Multibit Σ–Δ A/D converter incorporating a novel class of dynamic element matching techniques," *IEEE Trans. Circuits Syst. II*, vol. 39, no. 1, pp. 35-51, Jan. 1992.

[6] R. T. Baird and T. S. Fiez, "Linearity enhancement of multibit $\Delta\Sigma$ A/D and D/A converters using data weighted averaging," *IEEE Trans. Circuits Syst. II*, vol. 42, no. 12, pp. 753-762, Dec. 1995.

[7] T. Kwan, R. Adams, and R. Libert, "A stereo multibit SD DAC with asynchronous master-clock interface," *IEEE J. Solid-State Circuits*, vol. 31, no. 12, pp. 1881-1887, Dec. 1996.

[8] R. K. Henderson and O. J. A. P. Nys, "Dynamic element matching techniques with arbitrary noise shaping function," in *Proc. ISCAS*, May 1996, pp. 293-296.

[9] X-M Gong *et al.*, "A 120 dB multi-bit SC audio DAC with second-order noise shaping," in *ISSCC Dig. Tech. Papers*, Feb. 2000, pp. 344-345.

[10] E. Fogleman, J. Welz, and I. Galton, "An Audio ADC delta-sigma modulator with 100-dB peak SINAD and 102-dB DR using a second-order mismatch-shaping DAC," *IEEE J. Solid-State Circuits*, vol. 36, no. 3, pp. 339-348, March 2001.

[11] M. Sarhang-Nejad and G. C. Temes, "A high resolution multibit $\Sigma\Delta$ ADC with digital correction and relaxed amplifier requirements," *IEEE J. Solid-State Circuits*, vol. 28, no. 6, pp. 648-660, Jun. 1993.

[12] R. T. Baird and T. S. Fiez, "A low oversampling ratio 14-b 500-kHz $\Delta\Sigma$ ADC with a self-calibrated multibit DAC," *IEEE J. Solid-State Circuits*, vol. 31, no. 3, pp. 312-320, March 1996.

[13] D.W. J. Groeneveld *et al.*, "A self-calibration technique for monolithic high-resolution D/A converters," *IEEE J. Solid-State Circuits*, vol. 24, no. 6, pp. 1517-1522, Dec. 1989.

[14] U-K Moon, J. Silva, J. Steensgaard, and G. C. Temes, "Switched-capacitor DAC with analogue mismatch correction," *IEE Electronics Lett.*, vol. 35, no. 22, pp. 1903-1904, Oct. 1999.

[15] C. Petrie and M. Miller, "A background calibration technique for multibit delta-sigma modulators," in *Proc. ISCAS*, May 2000, pp. 29-32.

[16] P. Kiss, U. Moon, J. Steensgaard, J. T. Stonick and G. C. Temes, "High-speed $\Sigma\Delta$ ADC with error correction," *IEE Electronics Lett.*, no. 2, pp. 76-77, Jan. 2001.

[17] X. Wang, P. Kiss, U. Moon, J. Steensgaard, and G. C. Temes, "Digital estimation and correction of DAC errors in multibit $\Delta\Sigma$ ADCs," *IEE Electronics Lett.*, vol. 37, no. 7, pp. 414-415, Mar. 2001.

[18] O. J. A. P. Nys and R. K. Henderson, "An analysis of dynamic element matching techniques in sigma-delta modulation," in *Proc. ISCAS*, May 1996, pp. 231-234.

[19] F. Chen and B. Leung, "Some observations on tone behavior in data weighted averaging," in *Proc. ISCAS*, May 1998, pp. 500-503.

[20] K-D Chen and T-H Kuo, "An improved technique for reducing baseband tones in sigma-delta employing data weighted averaging algorithms without adding dither," *IEEE Trans. Circuits Syst. II*, vol. 46, no. 1, pp. 63-68, Jan. 1999.

[21] M. Vadipour, "Techniques for preventing tonal behavior of data weighted averaging algorithm in Σ–Δ modulators," *IEEE Trans. Circuits Syst. II*, vol. 47, no. 11, pp. 1137-1144, Nov. 2000.

[22] I. Fujimori *et al.*, "A 90-dB SNR 2.5-MHz output-rate ADC using cascaded multibit delta-sigma modulation at 8x oversampling ratio," *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1820-1828, Dec. 2000.

[23] K. Vleugels, S. Rabii, and B. A. Wooley, "A 2.5-V sigma-delta modulator for broadband communications applications," *IEEE J. Solid-State Circuits*, no. 12, pp. 1887-1899, Dec. 2001

[24] R. E. Radke, A. Eshraghi, and T. S. Fiez, "A 14-bit current-mode $\Sigma\Delta$ DAC based upon rotated data weighted averaging," *IEEE J. Solid-State Circuits*, vol. 35, no. 8, pp. 1074-1084, Aug. 2000.

[25] Y. Geerts, M. S. J. Steyaert, and W. Sansen, "A high-performance multibit $\Delta\Sigma$ CMOS ADC," *IEEE J. Solid-State Circuits*, no. 12, pp. 1829-1840, Dec. 2000.