An Audio ADC Delta–Sigma Modulator with 100-dB Peak SINAD and 102-dB DR Using a Second-Order Mismatch-Shaping DAC

Eric Fogleman, *Member, IEEE*, Jared Welz, *Student Member, IEEE*, and Ian Galton, *Member, IEEE*

***Abstract*—A second-order audio analog-to-digital converter (ADC) $\Delta\Sigma$ modulator using a second-order 33-level tree-structured mismatch-shaping digital-to-analog converter (DAC) is presented. Key logic simplifications in the design of the mismatch-shaping DAC encoder are shown which yield the lowest complexity second-order mismatch-shaping DAC known to the authors. The phenomenon of signal-dependent DAC noise modulation in mismatch-shaping DACs is illustrated, and a modified second-order input-layer switching block is presented which reduces inband DAC noise modulation by 6 dB. Implementation details and measured performance of the 3.3-V 0.5-μm single-poly CMOS prototype are presented. All 12 prototype devices achieve better than 100-dB signal-to-noise-and-distortion and 102-dB dynamic range over a 10–20 kHz measurement bandwidth.**

***Index Terms*—analog–digital conversion, CMOS analog integrated circuits, delta–sigma modulation, digital–analog conversion, dynamic element matching, mixed analog–digital integrated circuits.**

I. Introduction

MULTIBIT $\Delta\Sigma$ modulation has recently been applied to implement high-performance $\Delta\Sigma$ analog-to-digital converters (ADCs) [1]–[7]. The use of multibit quantization in a $\Delta\Sigma$ modulator reduces the power of the quantization noise which must be shaped out of band. Thus, a multibit $\Delta\Sigma$ modulator can achieve the same signal-to-quantization-noise ratio as a 1-bit $\Delta\Sigma$ modulator with a lower modulator order and a reduced oversampling ratio. These benefits can be used to relax the performance requirements on the analog switched-capacitor circuitry and enable the implementation of high-performance $\Delta\Sigma$ ADCs in CMOS fabrication processes optimized for digital circuits [7]. However, realizing these benefits requires an internal digital-to-analog converter (DAC) with linearity and inband noise specifications equal to or better than those of the $\Delta\Sigma$ ADC. The development of mismatch-shaping DACs has helped to address this issue and make the implementation of multibit $\Delta\Sigma$ ADCs practical.

Manuscript received July 20, 2000; revised October 13, 2000. This work was supported by the National Science Foundation under Grant MIP-9711331 and by the Office of Naval Research under Grant N00014-98-1-0830.

E. Fogleman was with the Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093 USA. He is now with Broadcom, Irvine, CA 92619-7013 USA (e-mail: fogleman@broadcom.com).

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

Publisher Item Identifier S 0018-9200(01)01430-5.

Though second-order mismatch shaping algorithms theoretically offer increased inband precision over first-order algorithms, they present several challenges. The performance improvement gained by using second-order mismatch-shaping must be sufficient to justify its increased digital logic complexity. The encoder's switching or selection logic must be stable for real-world input signals. Finally, the encoder should minimize spurious tones and signal-dependent modulation of the DAC mismatch noise because these effects can limit the dynamic range of high-resolution $\Delta\Sigma$ ADCs. While several mismatch-shaping algorithms have been reported that provide first-order spectral shaping of static DAC mismatch errors [8]–[16], only a few provide second-order mismatch shaping [12]–[16]. Though [12]–[15] show simulated second-order mismatch shaping, detailed hardware-level implementations are not presented.

This paper presents a second-order audio ADC $\Delta\Sigma$ modulator using a second-order mismatch-shaping DAC based on the tree-structure presented in [15] that addresses these challenges. The prototype $\Delta\Sigma$ modulator uses the analog front end presented in [7] with a redesigned mismatch-shaping DAC encoder. The design presented here is the lowest complexity second-order mismatch-shaping DAC encoder known to the authors. The architecture of the mismatch-shaping DAC encoder permits a gate count reduction at the expense of a slight degradation in spectral shaping performance near dc. Thus in applications where analog circuit noise is dominant at low frequencies, significant hardware savings are possible. The tree-structured encoder employs a modified input-layer switching block which reduces signal-dependent DAC noise modulation by 6 dB, thereby improving the signal-to-noise-ratio (SNR) for low-level input signals. The prototype $\Delta\Sigma$ modulator is implemented in a standard 0.5-μm 3.3-V single-poly CMOS fabrication process. All 12 of the fabricated prototypes achieve a 100-dB peak signal-to-noise and distortion ratio (SINAD) and 102-dB dynamic range over a 10–20 kHz measurement bandwidth.

II. Mismatch-Shaping DAC Signal Processing

The prototype uses the second-order $\Delta\Sigma$ modulator architecture shown in Fig. 1 with 33-level quantization and feedback [18]. As mentioned in Section I, the prototype's switched-capacitor analog front end and flash ADC are identical to those presented in [7]. Because the dynamic range of the $\Delta\Sigma$ modulator in [7] was limited by the first-order shaped DAC mismatch

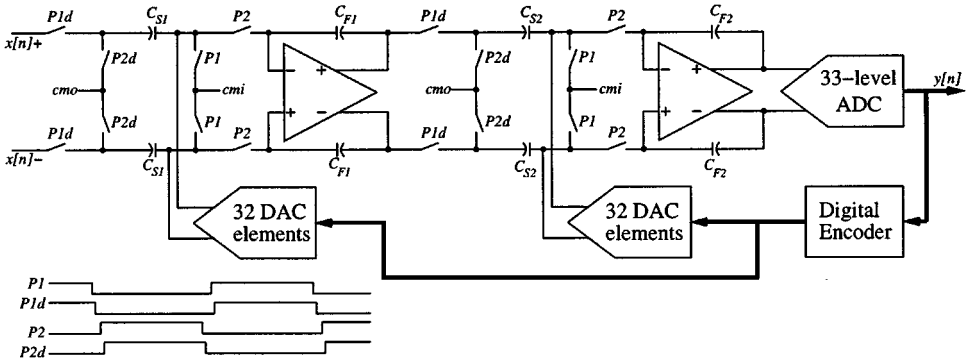

Fig. 1. Switched-capacitor implementation of the $\Delta\Sigma$ modulator.

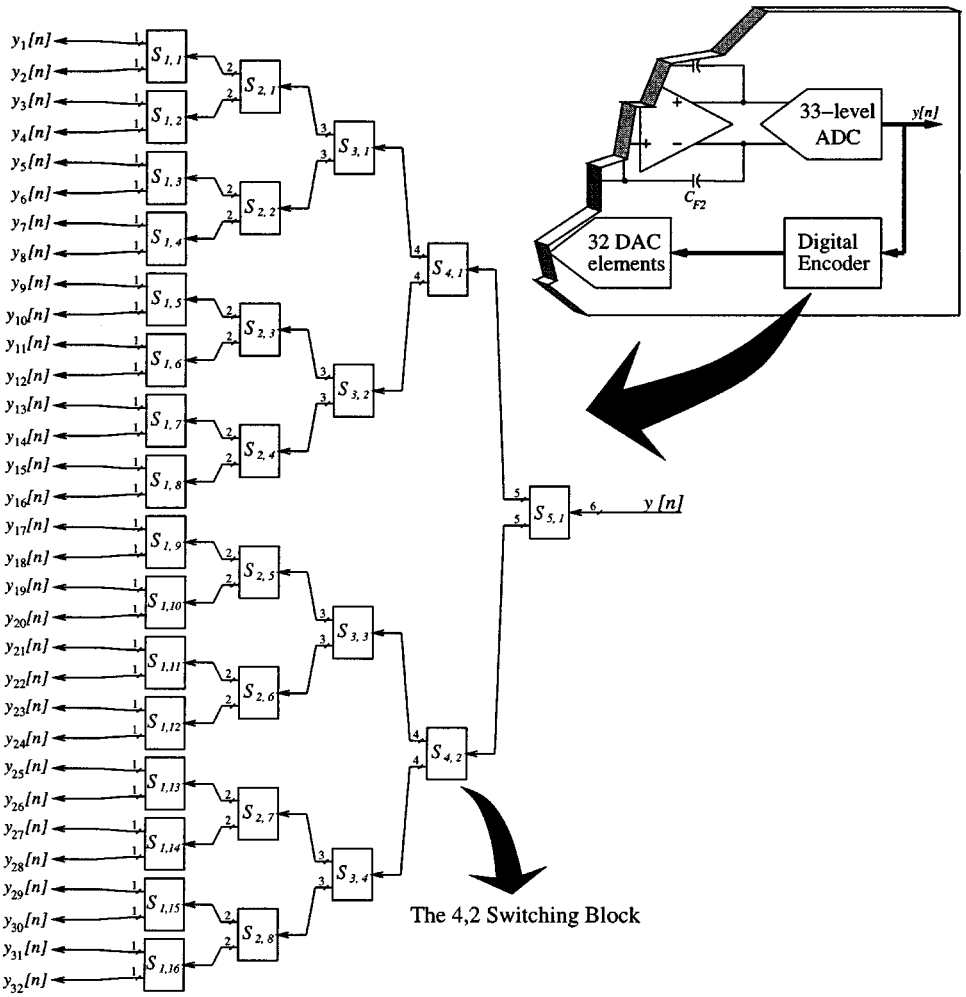

Fig. 2. The 33-level tree-structured mismatch-shaping DAC encoder.

noise at the upper end of the passband, the goal of this prototype was to use second-order mismatch-shaping to extend the $\Delta\Sigma$ modulator's dynamic range without modifying the analog circuitry or significantly increasing the die area.

The high-level architecture of the prototype's 33-level mismatch-shaping DAC encoder is shown in Fig. 2. The tree-structured encoder, based on the architecture presented in [15], generates 32 element selection lines that control the $\Delta\Sigma$ modulator's two unit-element DAC arrays. Each column of the tree structure in Fig. 2 is called a *layer*. The layers are labeled 5 through 1, where *layer 5* is the input layer and *layer 1* is the output layer.

The elements within the tree structure are called *switching blocks* and are labeled $S_{k,r}$, where k refers to the encoder layer and r refers to the position within the layer. The switching block $S_{k,r}$ has a $(k+1)$-bit input, $y_{k,r}[n]$, and two k-bit output sequences, $y_{k-1,2r-1}[n]$ and $y_{k-1,2r}[n]$. For example, the $S_{4,2}$ switching block in Fig. 2 has a 5-bit input, $y_{4,2}[n]$, and two 4-bit outputs, $y_{3,3}[n]$ and $y_{3,4}[n]$. To simplify the notation in Fig. 2,

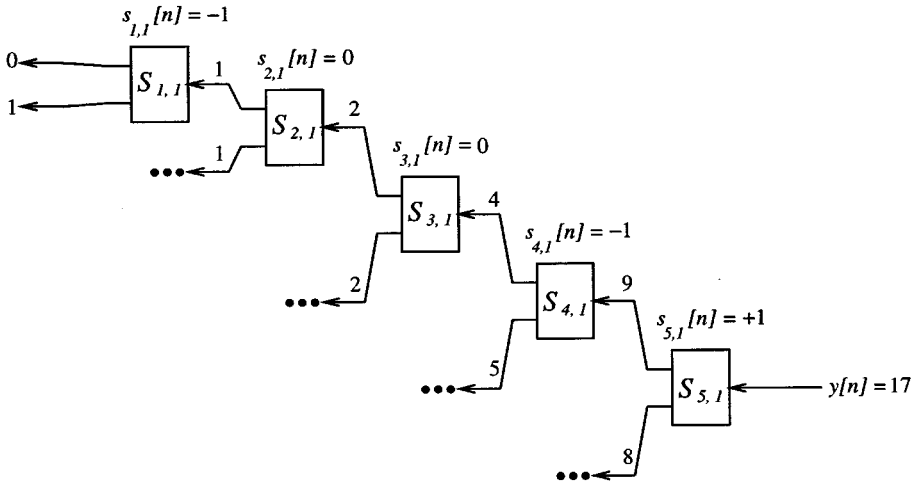

Fig. 3. Data flow along a single branch of the mismatch-shaping DAC encoder illustrating how the switching sequences $s_{k,r}[n]$ control the propagation of data to the output layer.

the encoder's input $y_{5,1}[n]$ is denoted by $y[n]$ and the encoder's outputs $y_{0,r}[n]$ are denoted by $y_r[n]$.

The mismatch-shaping DAC encoder converts its binary-coded 33-level input value into an output value encoded on the 32 DAC element selection lines. The encoder provides spectral shaping of DAC mismatch errors while operating under the constraint that the number of DAC element selection lines asserted at its output equals the binary value at its input. To satisfy this constraint, each switching block is required to operate such that the sum of its two outputs equals its input, and that the magnitude of each output is less than or equal to the number of DAC elements that it controls. That is, for each $S_{k,r}$

$$y_{k-1,2r-1}[n] + y_{k-1,2r}[n] = y_{k,r}[n] \tag{1}$$

and

$$0 \le y_{k,r}[n] \le 2^k. \tag{2}$$

Equations (1) and (2) constitute the *number conservation rule* described in [15].

Each switching block $S_{k,r}$ has an associated *switching sequence*, $s_{k,r}[n]$, which is defined as the difference between its outputs

$$s_{k,r}[n] = y_{k-1,2r-1}[n] - y_{k-1,2r}[n]. \tag{3}$$

From (1) and (3), it follows that the $S_{k,r}$ switching block implements

$$y_{k-1,2r-1}[n] = \frac{1}{2}(y_{k,r}[n] + s_{k,r}[n]) \tag{4}$$

and

$$y_{k-1,2r}[n] = \frac{1}{2}(y_{k,r}[n] - s_{k,r}[n]). \tag{5}$$

Thus, the switching sequence $s_{k,r}[n]$ determines how a switching block divides its input between its two outputs.

It is shown in [15] that the switching sequences determine the spectral shaping of the DAC noise; the tree-structured mismatch-shaping DAC's output voltage $v[n]$ is the input sequence $y[n]$ multiplied by a constant gain factor plus a dc offset and *DAC noise*, where the DAC noise is given by

$$e[n] = \sum_{k=1}^{5} \sum_{r=1}^{2^{5-k}} \Delta_{k,r} s_{k,r}[n].$$

The $\Delta_{k,r}$ coefficients are constants which depend only on the static DAC mismatches. Therefore, if the switching sequences are all uncorrelated and have the same spectral characteristics (e.g., second-order highpass spectral shaping), then the DAC noise will also possess this spectral shaping. Therefore, the task of designing a tree-structured mismatch-shaping DAC encoder amounts to designing a switching sequence generator which obeys (1) and (2) and provides the desired spectral shaping.

It can be verified that the following restriction on $s_{k,r}[n]$ is sufficient to satisfy (1) and (2):

$$s_{k,r}[n] = \begin{cases} 0, & \text{if } y_{k,r}[n] \text{ even} \\ \pm 1, & \text{if } y_{k,r}[n] \text{ odd.} \end{cases} \tag{6}$$

Thus, it follows from (4) and (5) that even switching block inputs are divided equally and distributed to the switching block's outputs and that odd switching block inputs are divided into values which differ by one. For odd inputs, the polarity of the switching sequence determines how the unequal values are distributed to the switching block's outputs.

Fig. 3 illustrates the operation of the switching blocks along a single branch of the encoder. The switching blocks with odd inputs—$S_{5,1}, S_{4,1}$, and $S_{1,1}$—have $|s_{k,r}[n]| = 1$, while those with even inputs—$S_{3,1}$ and $S_{2,1}$—have $s_{k,r}[n] = 0$. As noted above, the sign of $s_{k,r}[n]$ determines how odd inputs are distributed to the outputs of the switching block. For example, $s_{5,1}[n] = 1$ sends 9 to the upper output and 8 to the lower output of $S_{5,1}$, while $s_{4,1}[n] = -1$ sends 4 to the upper output and 5 to the lower output of $S_{4,1}$.

A. Second-Order Switching Sequence Generator

To achieve second-order spectral shaping given (6), each switching sequence generator must produce a sequence $s_{k,r}[n]$,

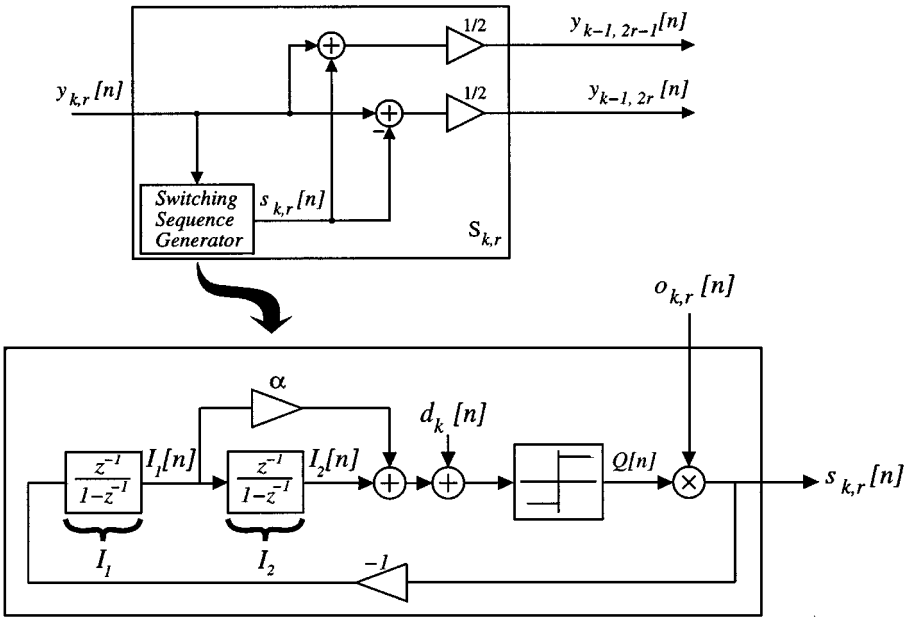

Fig. 4. Second-order switching block signal processing.

taking on values -1, 0, and $+1$ which has a power spectral density (PSD) with a 12-dB/octave slope near dc. If it were not for the number conservation rule, a standard second-order digital $\Delta\Sigma$ modulator with a zero input could be used, but (6) requires that $s_{k,r}[n] = 0$ whenever the input to $S_{k,r}$ is even.

Instead, the modified second-order digital $\Delta\Sigma$ modulator shown in Fig. 4 could be used as the switching sequence generator.[1] The $\Delta\Sigma$ modulator's first and second integrators are denoted by I_1 and I_2, respectively. The sequence $o_{k,r}[n]$ represents the parity of switching block's input. Thus, (6) can be rewritten as

$$s_{k,r}[n] = \begin{cases} 0, & \text{if } o_{k,r}[n] = 0 \\ \pm 1, & \text{if } o_{k,r}[n] = 1. \end{cases} \tag{7}$$

If $y_{k,r}[n]$ is represented as a binary number, $o_{k,r}[n]$ is the least significant bit (LSB) of $y_{k,r}[n]$. The multiplier, shown after the comparator in Fig. 4, ensures that (7) is satisfied, and the comparator limits $|s_{k,r}[n]| \leq 1$. The dither sequence $d_k[n]$ is a sequence of i.i.d. uniform random variables on $(-1, 1)$ that is used to reduce spurious tones in $s_{k,r}[n]$. The feedforward gain element, α, in Fig. 4 improves the stability of the sequence generator by providing a means for I_1 to influence $s_{k,r}[n]$ when I_2 takes on large positive or negative values.

Unlike the first-order switching blocks in [7] and [15], the second-order switching sequence generator in Fig. 4 can exhibit instability. For example, if the first integrator's output $I_1[n]$ is nonzero and a sequence of even switching block inputs occurs, the second integrator's output $I_2[n]$ will continue to grow in magnitude until an odd input occurs. Unless restrictions are placed on the occurrences of even values of $y_{k,r}[n]$, no bound can be placed on $|I_2[n]|$. Therefore, the number of bits required to represent $I_1[n]$ and $I_2[n]$ depends on the choice of α and the statistics of $y_{k,r}[n]$.

The switching sequence generator in Fig. 4 can be modified to address this issue and reduce its hardware complexity by utilizing the fact that $s_{k,r}[n]$ retains second-order spectral shaping for large values of α. Thus, if I_1 and I_2 are designed to saturate at some magnitude, M (i.e., $|I_1[n]| \leq M$ and $|I_2[n]| \leq M$), a gain of $\alpha = M + 1$ can be implemented by allowing $I_1[n]$ to "override" $I_2[n]$ whenever $I_1[n]$ is nonzero rather than explicitly computing $\alpha I_1[n] + I_2[n] + d_k[n]$. Choosing integer-valued initial conditions for I_1 and I_2 guarantees that $I_1[n]$ and $I_2[n]$ will take on integer values for all n. Thus, a sequence of i.i.d. random variables with $P(d_k[n] = -1/2) = P(d_k[n] = +1/2) = 1/2$ is sufficient to dither the sequence generator.

To implement Fig. 4 in digital hardware, the three-valued switching sequence $s_{k,r}[n]$ must be represented as a two-bit quantity. From (7), it follows that $|s_{k,r}[n]| = o_{k,r}[n]$. By defining

$$q_{k,r}[n] = \begin{cases} 1, & \text{if } s_{k,r} > 0 \\ 0, & \text{if } s_{k,r} < 0 \end{cases}$$

$q_{k,r}[n]$ and $o_{k,r}[n]$ form a sign/magnitude representation of $s_{k,r}[n]$.

With the sign/magnitude representation of $s_{k,r}[n]$, the switching sequence generator in Fig. 4 can be implemented in hardware as shown in Fig. 5. The integrators, I_1 and I_2, are implemented as up/down counters with sign/magnitude outputs. The gain element, adders, and comparator in Fig. 4 have been replaced by the decision logic in Table I, where $r_k[n]$ is a sequence of i.i.d. random variables with $P(r_k[n] = 1) = P(r_k[n] = 0) = (1/2)$. The sequence $r_k[n]$ is the hardware realization of $d_k[n]$ in Fig. 4. Provided $I_1[0] = 0$, the feedback ensures that $I_1[n]$ is limited to values in the set $\{-1, 0, 1\}$ for $n > 0$. Thus, a two-bit counter is sufficient to implement I_1. The b-bit I_2 counter is designed such that it

[1]This switching sequence generator with a feedforward gain of 8 and no dither was used to generate the simulation results shown in [14] though the circuit itself was not published.

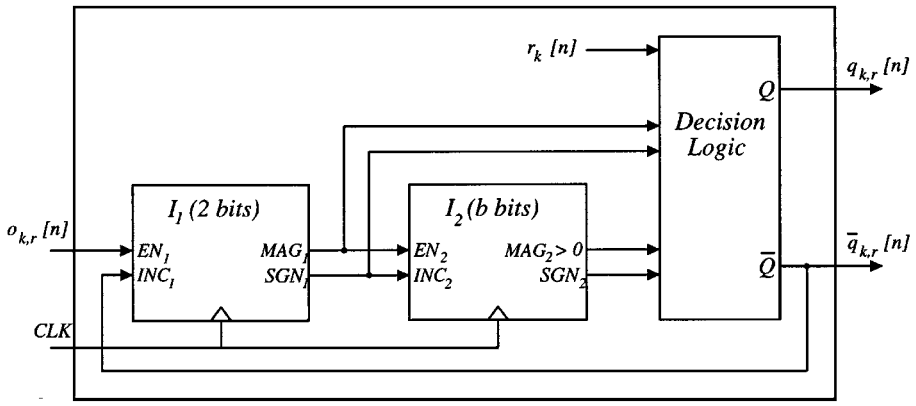

Fig. 5. Logic implementation of the second-order switching sequence generator.

TABLE I
TRUTH TABLE FOR SWITCHING SEQUENCE GENERATOR DECISION LOGIC; "+", "−", AND "x" INDICATE POSITIVE, NEGATIVE, AND DON'T CARE VALUES, RESPECTIVELY, FOR $I_1[n]$ AND $I_2[n]$

$I_1[n]$	$I_2[n]$	$q_{k,r}[n]$
−	x	0
0	−	0
0	0	$r_k[n]$
0	+	1
+	x	1

saturates at its extreme values rather than "rolling over" in a modulo fashion.

The decision logic in Table I ensures that I_1 controls $s_{k,r}[n]$ when I_2 saturates. Thus, in this overload condition, the switching sequence generator degrades to first-order spectral shaping until I_2 recovers from saturation. Because of this benign overload behavior, the bit width of I_2 can be reduced at the expense of minor degradation of the spectral shaping near dc. Fig. 6 shows the DAC noise PSD for various I_2 bit widths. With 6-bit I_2 counters under the conditions simulated, the DAC encoder's switching sequence generators do not saturate and the DAC noise shows ideal second-order shaping. When the I_2 counters are reduced to 4 bits, the switching sequence generators saturate and the PSD changes from a 12-dB/octave slope to a 6-dB/octave slope near dc. For an encoder with 2-bit I_2 counters, saturation occurs frequently. The PSD has a 6-dB/octave slope for most frequencies, but the inband DAC noise power is approximately 10 dB lower than that of the first-order encoder. Even though the encoder with 2-bit I_2 counters does not provide true second-order shaping, it offers a substantial performance improvement over the first-order encoder with little additional hardware.

B. Signal-Dependent DAC Noise Modulation

As implemented in Figs. 4 and 5, $s_{k,r}[n]$ is forced to zero when $y_{k,r}[n]$ is even. As a result, long sequences of even inputs to a switching block will affect the spectral shaping of $s_{k,r}[n]$. Fig. 7 shows the PSDs of individual $s_{k,r}[n]$ sequences where the parity of the switching block's input $o_{k,r}[n]$ is a sequence of i.i.d. random variables with $P(o_{k,r}[n] = 1) = p$ and $P(o_{k,r}[n] = 0) = 1 - p$, and where the sizes of the I_2 counters are sufficiently large to avoid overload. As the probability

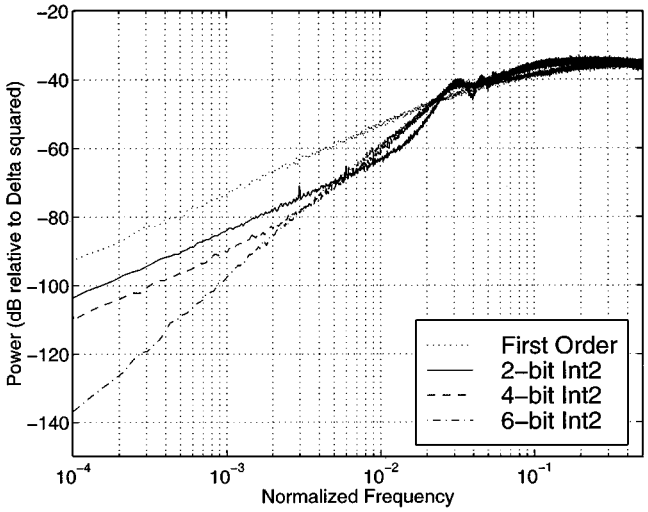

Fig. 6. Simulated DAC noise PSDs for a 33-level mismatch-shaping DAC; first-order and second-order shaping.

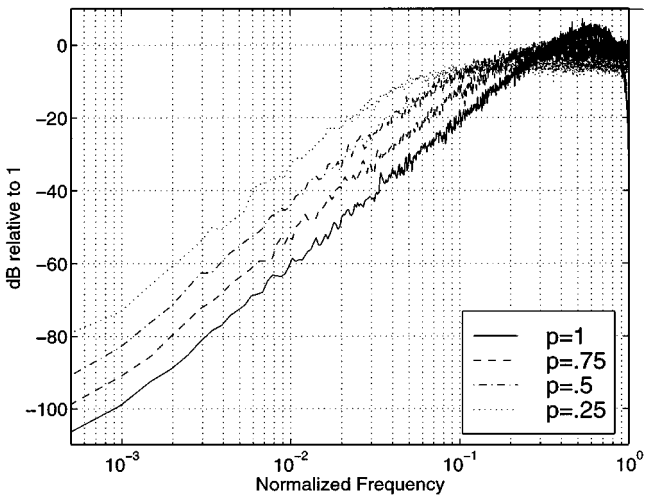

Fig. 7. Simulated PSDs of individual $s_{k,r}[n]$ sequences for several values of $P(o_{k,r} = 1)$.

of even inputs increases, the inband noise power rises dramatically. It can be seen in Fig. 7 that while all of the PSDs retain a 12-dB/octave slope near dc, the inband noise power increases by 20 dB as $P(o_{k,r}[n] = 1)$ goes from 0.75 to 0.25. Because

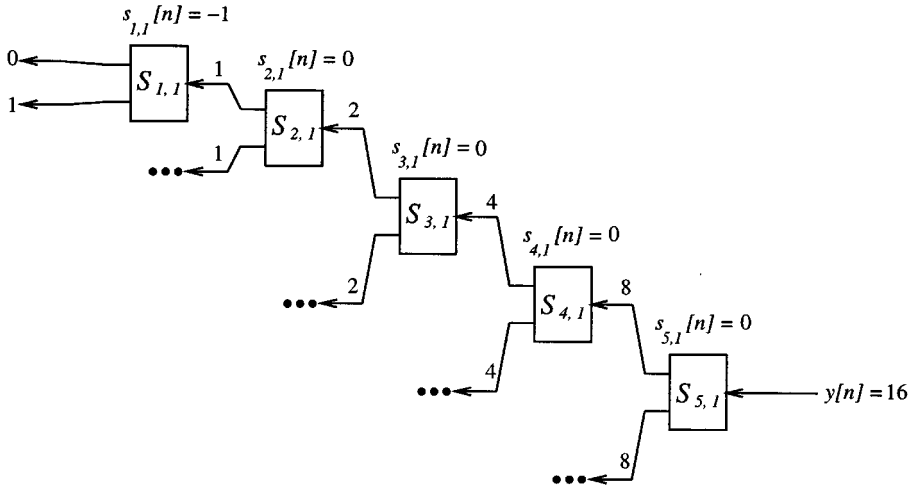

Fig. 8. Data flow along a single branch of the mismatch-shaping DAC encoder illustrating how an input of 16 results forces $s_{k,r}[n] = 0$ for all but the output layer.

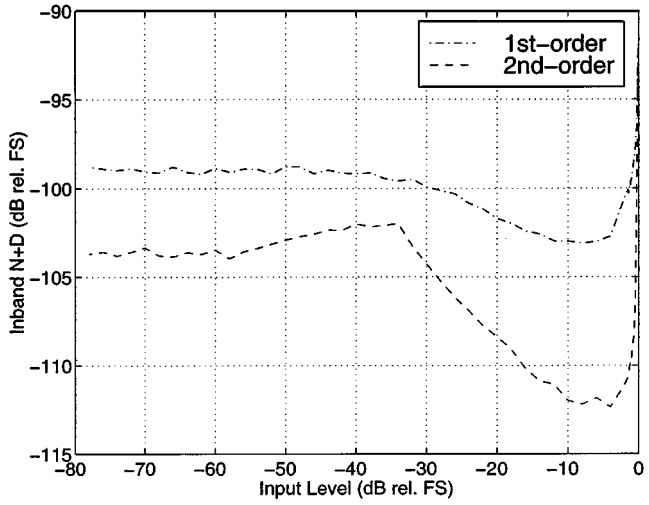

Fig. 9. Simulated inband noise versus input level for the $\Delta\Sigma$ modulator using first-order and second-order mismatch-shaping DAC encoders.

the DAC noise is the weighted sum of the $s_{k,r}[n]$ sequences, this directly translates to increased inband DAC noise.

For the tree structure of Fig. 2, input values that are powers of two pose a particular problem because they propagate even inputs to successive layers of the tree structure. A commonly encountered input for the 33-level DAC encoder is an input of 16, the $\Delta\Sigma$ modulator's midscale value. Fig. 8 illustrates how an input of 16 is factored into successive even values until the last layer of the encoder. As a result, 15 of the 31 $s_{k,r}[n]$ sequences have elevated inband noise power as illustrated in Fig. 7.

The dashed-line plot in Fig. 9 shows simulated inband noise power versus input level for the $\Delta\Sigma$ modulator of Fig. 1 using the second-order mismatch-shaping encoder shown in Figs. 2 and 5 with 4-bit I_2 counters and $3\sigma = 1\%$ Gaussian DAC element mismatches. For this implementation, the inband noise power increases by 10 dB as the input drops from -4 to -35 dB. This is a direct result of the DAC encoder receiving a high density of midscale inputs for low-level $\Delta\Sigma$ modulator inputs. Combined with the inband thermal noise and $1/f$ noise of the analog front end, this level of DAC noise modulation would have limited the $\Delta\Sigma$ modulator's dynamic range to 100 dB. Thus, for low-level $\Delta\Sigma$ modulator input signals, second-order mismatch shaping would have provided little or no improvement in SNR over first-order mismatch shaping.

This effect can be mitigated by noting that (7) is sufficient but not necessary to satisfy the number conservation rule. For example, the $S_{5,1}$ block is free to set $s_{5,1}[n] = \pm 2$ for all inputs divisible by four except 0 and 32 (i.e., $y_{5,1}[n] \in \{4, 8, \ldots 28\}$). Thus, when $S_{5,1}$ receives a high density of inputs with value 16, it can still produce a second-order shaped $s_{5,1}[n]$ taking on values ± 2. With this modification, the $S_{5,1}$ block can break the midscale input into output values of 7 and 9, as illustrated in Fig. 10, and prevent the propagation of even values to the remaining layers of the encoder.

Fig. 11 shows the modifications required for the $S_{5,1}$ switching sequence generator. As in Fig. 4, it produces $s_{5,1}[n] = \pm 1$ for $o_{5,1}[n] = 1$. The gain element and $f_{5,1}[n]$ multiplier following the comparator are used to produce $s_{5,1}[n] = \pm 2$ for inputs divisible by four other than the full-scale values. The sequence $f_{5,1}[n]$ is defined as

$$f_{5,1}[n] = \begin{cases} 1, & \text{if 4 divides } y_{5,1}[n] \quad \text{and} \quad y_{5,1}[n] \neq 0, 32 \\ 0, & \text{otherwise.} \end{cases}$$

When $y_{5,1}[n]$ is represented as a binary number, divisibility by four can be determined by testing if the two least significant bits of $y_{5,1}[n]$ are zero. Full-scale inputs can be detected by testing if all bits of $y_{5,1}[n]$ are low or high.

Fig. 12 shows a comparison between the simulated inband noise and distortion of the $\Delta\Sigma$ modulator using the modified second-order mismatch shaping DAC encoder and those illustrated previously in Fig. 9. The solid-line plot shows that the modified second-order encoder reduces the DAC noise modulation by 6 dB and provides significantly better performance than the first-order encoder over the entire range of input amplitudes. Though this modification roughly doubles the gate count of the $S_{5,1}$ block, the encoder's overall gate count does not increase significantly because the other 30 switching blocks use

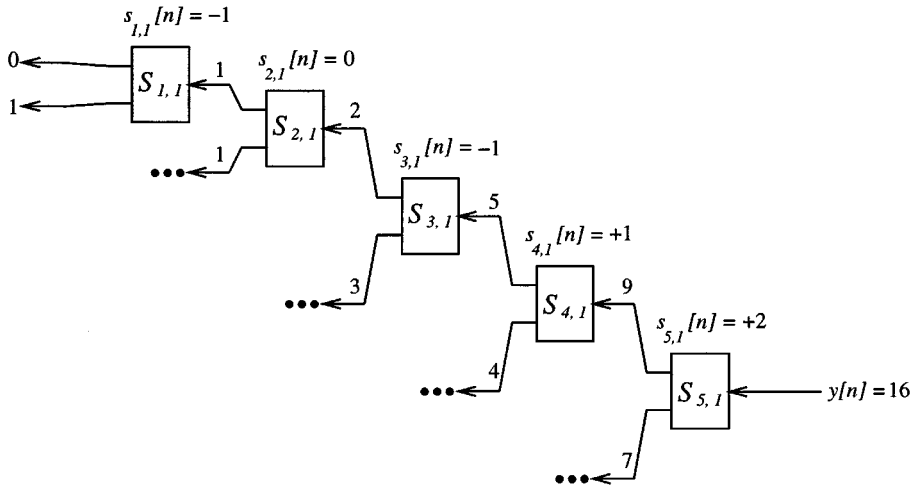

Fig. 10. Data flow along a single branch of the mismatch-shaping DAC encoder illustrating how the modified $S_{5,1}$ block prevents the propagation of even values to successive layers.

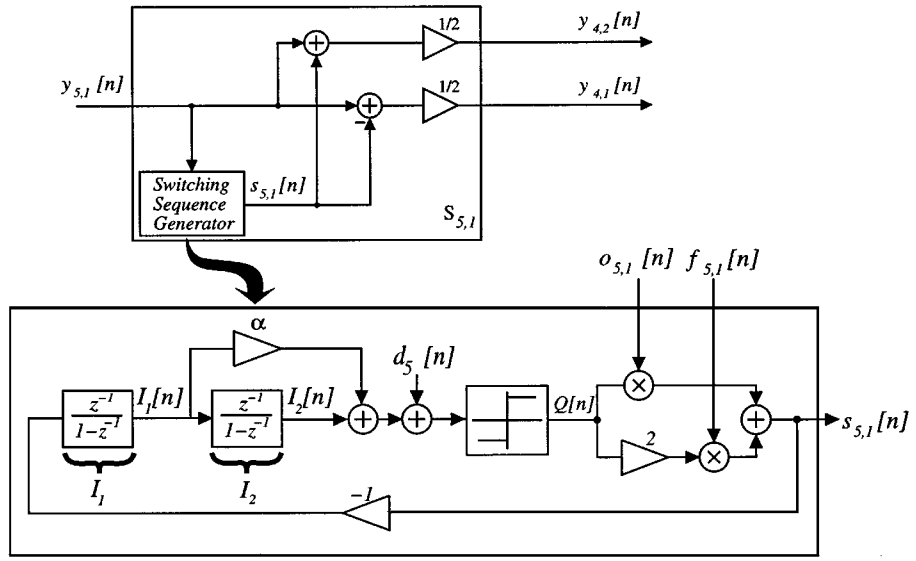

Fig. 11. Signal processing of the modified $S_{5,1}$ switching block which allows ± 1 feedback for odd inputs and ± 2 feedback for inputs divisible by four.

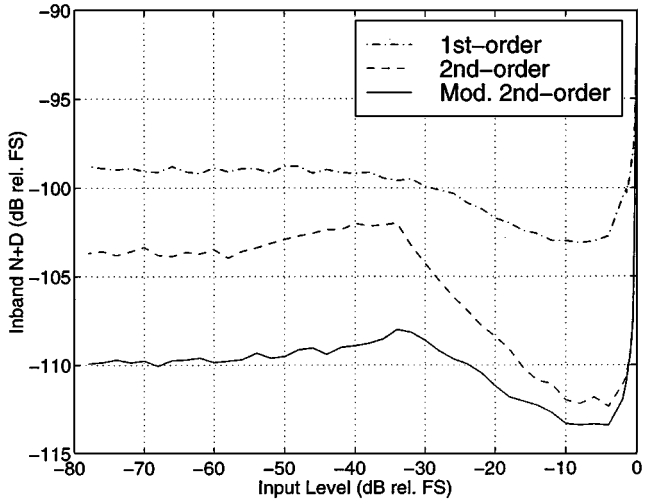

Fig. 12. Simulated inband noise versus input level for the $\Delta\Sigma$ modulator using first-order, second-order, and modified second-order mismatch-shaping DAC encoders.

the second-order switching sequence generator shown in Fig. 5.

III. Prototype Results

The prototype $\Delta\Sigma$ modulator IC shown in Fig. 13 was fabricated in a 0.5-μm 3.3-V triple-metal single-poly CMOS process. The analog front end of the prototype—the switched-capacitor circuitry and flash ADCs with comparator offset DEM and digital common mode rejection (DCMR)—are identical to the audio ADC $\Delta\Sigma$ modulator using a first-order mismatch shaping DAC presented in [7]. Logic reduction and layout optimization were performed on the DCMR logic to reduce its area. Though the second-order mismatch-shaping DAC encoder with the modified $S_{5,1}$ block is approximately twice the die area of the first-order encoder of [7], the area reduction of the DCMR logic permitted the second-order design to fit in same pad ring as the first-order design.

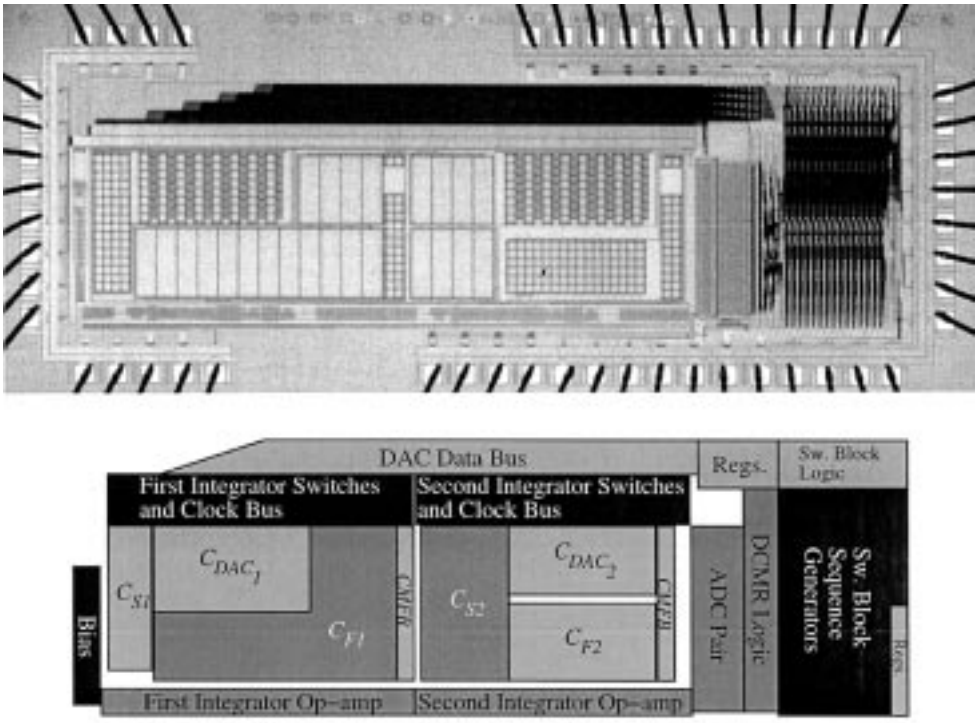

Fig. 13. Prototype $\Delta\Sigma$ modulator die photograph and layout floorplan.

TABLE II
PERFORMANCE AND SPECIFICATION SUMMARY

Sample Rate	3.072 MHz
Oversampling Ratio	64
Full-scale input range	±3.0 V peak differential
Peak SINAD	100.3 dB at 2.8 V, 10 Hz — 20 kHz
SFDR	105.6 dB, 10 Hz — 20 kHz
DR	102.5 dB, 10 Hz — 20 kHz
Power dissipation	66 mW analog, 4.4 mW digital
Technology	3.3-V, 0.5-μm CMOS (1 poly, 3 metal)
Chip size	5 mm × 1.9 mm
Package	65 pin PGA

Each second-order switching sequence generator was implemented with a four-bit second integrator to save hardware in the mismatch-shaping DAC encoder. As described in Section II, this leads to occasional saturation within the switching sequence generators and causes the DAC noise to change from a second-order slope to a first-order slope near dc as shown in Fig. 6. Nevertheless, this design still provides significantly improved suppression of inband DAC noise relative to the first-order mismatch-shaping DAC in [7]. The switching block with the switching sequence generator shown in Fig. 5 required 31 gates, while the modified $S_{5,1}$ block in Fig. 11 required 58 gates. Thus the entire second-order mismatch-shaping DAC encoder was implemented using 988 gates. This is 28% of the gate count reported in [17] for a nine-level second-order mismatch-shaping DAC encoder.

Table II summarizes the device specifications and worst case measured performance for the 12 fabricated prototypes. Comparing worst case performance for both prototypes over a 10–20 kHz bandwidth, the $\Delta\Sigma$ modulator presented here has a 1.5-dB greater peak SINAD and a 2.3-dB greater dynamic range (DR) than the design presented in [7] with no increase in die area and only a 2.6% increase in power dissipation. Fig. 14 shows the measured 1.5-kHz SINAD versus input level. Fig. 15 shows the measured inband PSD for −1- and −60-dB input signals.

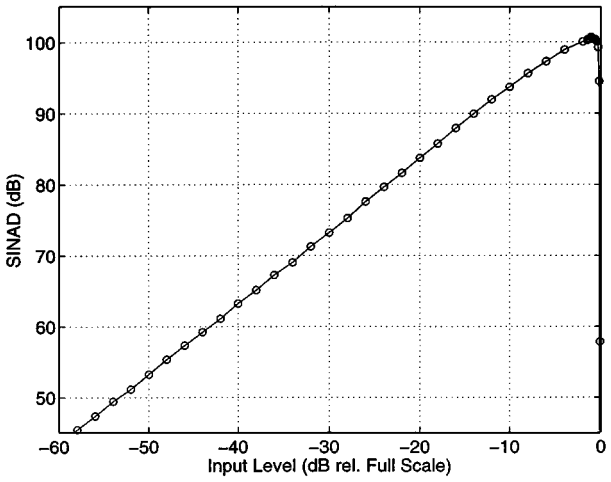

Fig. 14. Measured SINAD versus input level for the prototype $\Delta\Sigma$ modulator.

Fig. 16 shows the measured inband noise and distortion versus input level for the prototype $\Delta\Sigma$ modulator using the second-order mismatch-shaping DAC with the modified input-layer switching block. Performance for the $\Delta\Sigma$ modulator presented in [7] with a first-order mismatch-shaping DAC is included for comparison. The use of second-order mismatch shaping eliminates DAC mismatch noise as a dominant noise source for full-scale input signals. The modification to the input-layer switching block ensures that the DAC noise power in the audio band remains well below the circuit noise for low-level input signals.

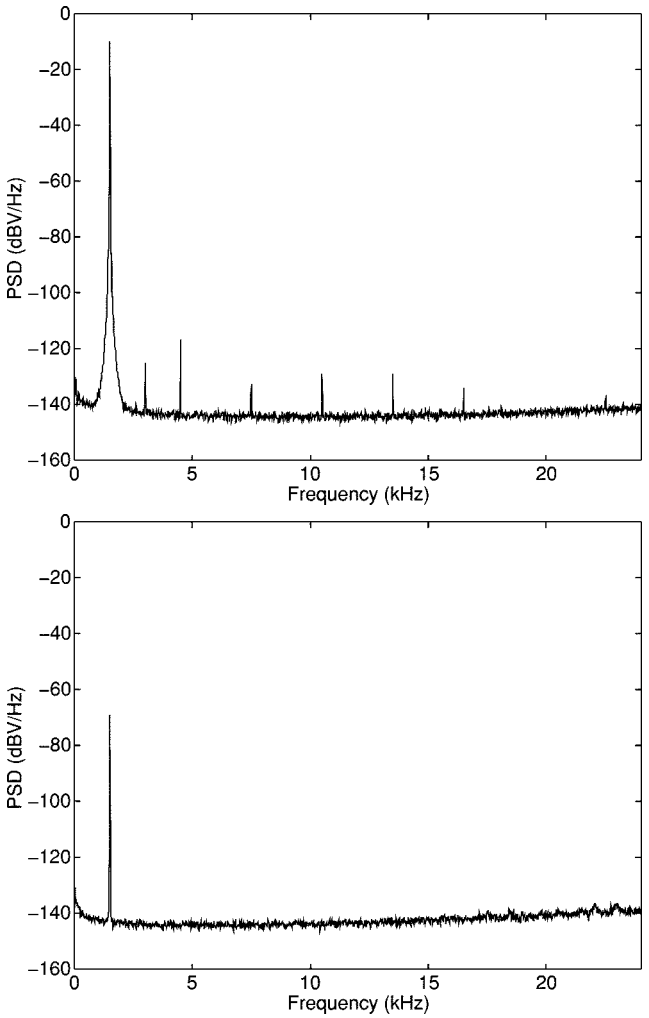

Fig. 15. Measured inband PSD for −1-dB and −60-dB, 1.5-kHz input signals.

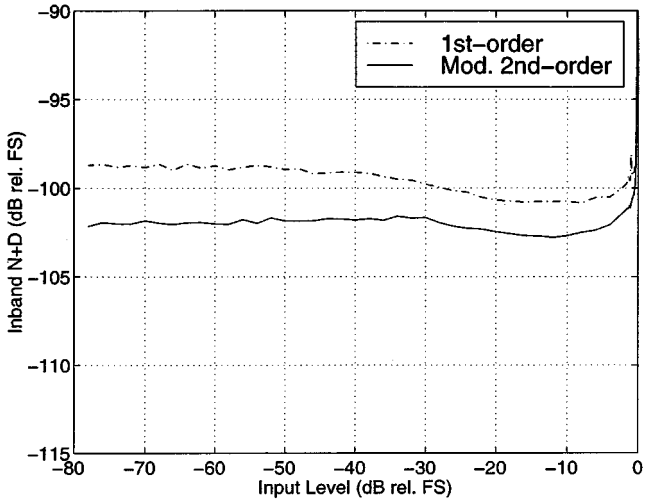

Fig. 16. Measured inband noise and distortion for the prototype $\Delta\Sigma$ modulators using first-order and modified second-order mismatch-shaping DAC encoders.

Because the performance of the second-order prototype is limited by the thermal noise of the analog front end, the improvement in dynamic range is limited to approximately 2.3 dB. By comparing the degree of noise modulation between the two prototypes, it can be seen that the DAC mismatch noise has been reduced well below the other noise sources in the second-order prototype.

IV. Conclusion

These results demonstrate that a practical second-order mismatch-shaping DAC can be implemented to yield improved performance over a first-order design with only a modest increase in hardware complexity. Using the second-order switching block presented, pure second-order spectral shaping of the DAC mismatches can be sacrificed to yield substantial hardware savings with minimal impact on performance. The prototype's dynamic range performance shows that the input-layer switching block can be modified to significantly reduce the amount of signal-dependent DAC noise modulation with minimal additional hardware.

References

[1] 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, pp. 648–660, June 1993.

[2] J. W. Fattaruso, S. Kiriaki, M. de Wit, and G. Warwar, "Self-calibration techniques for a second-order multibit sigma–delta modulator," *IEEE J. Solid-State Circuits*, vol. 28, pp. 1216–1223, Dec. 1993.

[3] 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, pp. 312–320, Mar. 1996.

[4] T. Brooks, D. Robertson, D. Kelly, A. DelMuro, and S. Harston, "A 16-b sigma–delta pipeline ADC with 2.5-MHz output data rate," *IEEE ISSCC Dig. Tech. Papers*, vol. 40, pp. 209–210, Feb. 1997.

[5] K. Leung, E. Swanson, K. Leung, and S. Zhu, "A 5-V 118-dB delta–sigma analog-to-digital converter for wideband digital audio," *IEEE ISSCC Dig. Tech. Papers*, vol. 40, pp. 218–219, Feb. 1997.

[6] 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 $8\times$ oversampling ratio," *IEEE ISSCC Dig. Tech. Papers*, vol. 43, pp. 338–339, Feb. 2000.

[7] 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, pp. 297–307, Mar. 2000.

[8] M. J. Story, "Digital to analogue converter adapted to select input sources based on a preselected algorithm once per cycle of a sampling signal,", Aug. 11, 1992.

[9] B. H. Leung and S. Sutarja, "Multibit sigma–delta A/D converter incorporating a novel class of dynamic element matching techniques," *IEEE Trans. Circuits Syst. II*, vol. 39, pp. 35–51, Jan. 1992.

[10] H. S. Jackson, "Circuit and method for cancelling nonlinearity error associated with component value mismatches in a data converter," U.S. Patent 5 221 926, June 22, 1993.

[11] R. W. Adams and T. W. Kwan, "Data-directed scrambler for multibit noise shaping D/A converters," U.S. Patent 5 404 142, Apr. 4, 1995.

[12] 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.

[13] R. K. Henderson and O. Nys, "Dynamic element matching techniques with arbitrary noise shaping function," *Proc. IEEE Int. Symp. Circuits and Systems*, vol. 1, pp. 293–296, May 1996.

[14] A. Keady and C. Lyden, "Tree structure for mismatch noise-shaping multibit DAC," *Electron. Lett.*, vol. 33, no. 17, pp. 1431–1432, Aug. 1997.

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

[16] X. Gong, E. Gaalaas, M. Alexander, D. Hester, E. Walburger, and J. Bian, "A 120-dB multibit SC audio DAC with second-order noise shaping," *IEEE ISSCC Dig. Tech. Papers*, vol. 43, pp. 344–345, Feb. 2000.

[17] 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.

[18] G. Lainey, R. Saintlaurens, and P. Senn, "Switched-capacitor second-order noise-shaping coder," *Electron. Lett.*, vol. 19, pp. 149–150, Feb. 1983.

Eric Fogleman (M'91) received the B.S. degree in electrical engineering from the University of Maryland, College Park, in 1990, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California, San Diego (UCSD), in 1998 and 2000, respectively.

From 1990 to 1993, he was with Analog Devices, Wilmington, MA, as a Product/Test Engineer for audio converter products. From 1993 to 1996, he was with Brooktree Corporation, San Diego, as a Product Engineer and Design Engineer for computer audio and multimedia graphics products. From 1996 to 2000, he was a Graduate Student Researcher at UCSD. In 2000, he joined Broadcom Corporation, Irvine, CA, as a Design Engineer. His interests include mixed-signal circuit design and signal processing.

Jared Welz (S'98) received the B.S.E.E. degree from the University of California, Irvine, in 1993 and the M.S.E.E. degree from University of California, Los Angeles, in 1994 with an emphasis in communication theory. He is currently working toward the Ph.D. degree at the University of California, San Diego.

Between 1994 and 1997, he was employed by AirTouch International (now Vodafone AirTouch), Pacific Bell Wireless, and L.A. Cellular (now AT&T Wireless). His 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.

He is currently an Associate Professor at the University of California, San Diego. He was formerly with the University of California, Irvine, Acuson, and Mead Data Central, and has acted as a regular Consultant for several companies. His research interests include integrated circuits and systems for communications including ADCs, DACs, frequency-to-digital converters, frequency synthesizers, synchronization and offset removal blocks for digital modems, and receiver linearization circuits. He holds four patents.