

# UC Davis

## UC Davis Previously Published Works

### Title

Background Adaptive Cancellation of Digital Switching Noise in a Pipelined Analog-to-Digital Converter Without Noise Sensors

### Permalink

<https://escholarship.org/uc/item/9gh1v873>

### Journal

IEEE Journal of Solid-State Circuits, 49(6)

### ISSN

0018-9200

### Authors

Chang, Nick C-J  
Hurst, Paul J  
Levy, Bernard C  
et al.

### Publication Date

2014-06-01

### DOI

10.1109/jssc.2014.2314446

Peer reviewed

# Background Adaptive Cancellation of Digital Switching Noise in a Pipelined Analog-to-Digital Converter Without Noise Sensors

Nick C.-J. Chang, Paul J. Hurst, *Fellow, IEEE*, Bernard C. Levy, and Stephen H. Lewis, *Fellow, IEEE*

**Abstract**—Switching noise generated by digital circuits can degrade analog circuit performance in mixed-signal integrated circuits (ICs). In an analog-to-digital converter (ADC), one major source of switching noise is digital output drivers. Traditional methods for mitigating this problem mostly have been to isolate the analog and digital circuits to minimize digital noise coupling into sensitive analog nodes. This paper presents two fully digital and adaptive algorithms, which find and cancel errors due to switching noise coupling at the output of an ADC without using noise sensors. To demonstrate the operation of these algorithms, a 12 bit, 40 MS/s pipelined ADC has been designed and fabricated in 0.18  $\mu\text{m}$  CMOS process. The system consists of an ADC with its own output drivers, and eight other independent digital output drivers (noise buffers) that can be programmed to produce four different kinds of switching noise. The switching noise cancellation (SNC) algorithm estimates noise parameters and stores them in look-up tables. At the ADC output, the effects of switching noise are digitally removed to recover the input samples. Test results show that the ADC achieves a signal-to-noise-and-distortion-ratio (SNDR) of 64.9 dB with all of the noise-generating buffers off. With the noise buffers on, the worst case SNDR before and after SNC is 51.9 dB and 63.7 dB, respectively.

**Index Terms**—additive noise, analog-to-digital converter, background adaptive noise cancellation, digital switching noise, least-mean-square estimation, LMS, maximum-likelihood estimation, multiplicative noise, noise-induced gain error pipelined ADC, power supply bounce, simultaneous switching noise, SSN, substrate noise.

## I. INTRODUCTION

**A**NALOG and digital circuits are routinely included together on integrated circuits, and the trend is toward full system integration on a single chip. The main potential advantage of this approach is that it can reduce cost and power dissipation. Unfortunately, integrating analog and digital circuits on

the same die degrades the raw performance of analog circuits because of switching noise coupling from the digital side [1]. In many mixed-signal IC designs, digital output drivers are a dominant source of switching noise because they are large in size and tend to switch simultaneously [2], [3]. Cases of switching noise degrading the performance of ADCs have also been reported [4]–[7].

Switching noise can couple to the analog signal path in several ways. For example, with a differential amplifier whose analog power supplies contain switching noise, an input-independent, additive noise component can appear in the differential output through device mismatch. Even with perfect matching, when the differential input is not zero, switching noise can appear at the output as a multiplicative noise component (i.e., the input is multiplied by scaled noise) [8] by coupling to the bias and modulating the small-signal parameters of the amplifier, such as the transconductance and output resistance or by varying the transistor's threshold voltage [9]. Also, with a differential sampling circuit that consists of transistor switches and sample capacitors, if switching noise is present in the substrate, this noise can couple to the circuit through parasitic capacitances that exist between the substrate and the top plates of the sample capacitors. If the differential paths are mismatched, an additive noise component appears at the output. Without the mismatch but with a nonzero differential input, a multiplicative noise component can appear at the output. The value of this multiplicative component depends on both the input and the noise [10] because the parasitic capacitance that links the substrate to the top plates of the sample capacitors contains junction capacitance that is input dependent.

A common method for mitigating the effects of switching noise has been to isolate the analog and digital circuits to minimize digital noise coupling into sensitive analog nodes. Isolation methods include large physical separation [11], the use of guard rings [12], [13], and the use of special process steps such as deep N-wells [2], [13]–[15]. While these methods can reduce the effects of switching noise coupling through the substrate, they increase die area and cost. Also, differential structures can reduce noise coupling effects by rejecting common-mode noise [3].

Finally, active noise minimization techniques have also been reported [9], [10], [16], [17]. In [10], negative feedback, substrate noise sensors, and dummy digital switches are used in a  $\Delta\Sigma$  loop to cancel substrate noise but are limited to use with

Manuscript received July 20, 2013; revised January 03, 2014, and March 05, 2014; accepted March 06, 2014. This paper was approved by Guest Editor Ichiro Fujimori. This work was supported in part by Analog Devices, Inphi, Linear Technology, Marvell Semiconductor, Northrop Grumman, the Agilent Foundation, the Broadcom Foundation, and NSF Grant 1016704. Chip fabrication was donated by TSMC.

N. C.-J. Chang was with the Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA, and is now with Marvell Semiconductor, Inc., Santa Clara, CA 95054 USA.

P. J. Hurst, B. C. Levy, and S. H. Lewis are with the Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA.

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

Digital Object Identifier 10.1109/JSSC.2014.2314446

TABLE I  
INVERTER OUTPUT  $w[m]$  TRANSITIONS AND THE CORRESPONDING  
NOISE VOLTAGE  $n[m]$

$w[m-1]$	$w[m]$	$n[m]$
0	0	0
0	1	$v_1$
1	0	$v_2$
1	1	0

$$|v_1| \neq |v_2| \neq 0$$

highly doped substrates. In [17], a spiral inductor feeds the detected noise signal on the digital power supply forward to an amplifier that produces an out-of-phase noise current to drive the substrate, but this technique is limited by the opamp bandwidth. To overcome these limitations, this paper presents a fully digital and background adaptive noise canceling technique that reduces the effects of switching noise coupling from digital buffers at the output of a pipelined ADC without noise sensors.

This paper is organized as follows. Section II briefly reviews the mechanism of switching noise generation. Section III shows a model of a pipelined ADC with switching noise. Section IV describes two algorithms that find and cancel the effects of switching noise due to digital output buffers. Section V presents the design of the prototype and its test modes. Then, measured results from the prototype chip are shown in Section VI, and finally, Section VII gives a conclusion.

## II. GENERATION OF SWITCHING NOISE

Standard CMOS logic has non-constant power supply and ground currents. With nonzero supply and ground inductances, these currents cause switching noise on the digital power supply and ground lines. This switching noise can inject into the substrate and often becomes the dominant source of switching noise [18]–[20]. Furthermore, if the power supply and ground are shared between analog and digital circuits, direct noise coupling occurs through the supplies.

Let  $n[m]$  denote the switching noise voltage sampled by an analog circuit at time index  $m$ . Table I shows the transitions of an inverter output  $w[m]$  and the corresponding  $n[m]$ , which is zero when the inverter's output is constant and nonzero otherwise. The magnitudes of the noise voltages  $v_1$  and  $v_2$  are not equal in general. Although every digital gate contributes switching noise, the largest gates contribute most of the noise in practice. In this work, the prototype includes large on-chip digital output buffers that are the dominant source of switching noise. For the remainder of this paper, these output buffers will be referred to as the *noise buffers*.

## III. PIPELINED ADC WITH SWITCHING NOISE

Fig. 1 shows a block diagram of a general pipelined ADC with  $K$  stages and with switching noise. As in a conventional pipelined ADC, each stage has an analog input and a low-resolution digital output  $D_1 \dots D_K$ . Also, each stage except the last produces an analog output  $r_1 \dots r_{K-1}$  that is the stage's amplified quantization error. However, unlike in a conventional

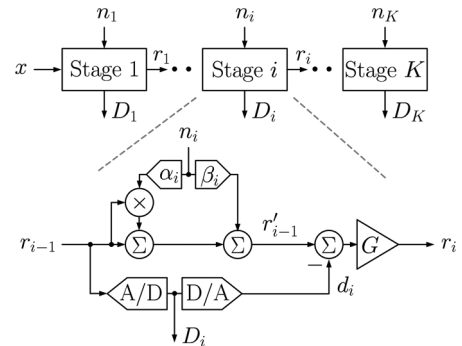


Fig. 1. Model of a pipelined ADC with switching noise. Scale factors  $\alpha_i$  and  $\beta_i$  are generally unknown. A multiplicative noise error is introduced by  $\alpha_i n_i$ , and an additive noise error is introduced by  $\beta_i n_i$ .

model, each stage also has a noise input  $n_1 \dots n_K$  to model the effect of undesired switching noise. An expanded view of stage  $i$  is also shown in Fig. 1.

As mentioned in Section I, switching noise  $n_i$  couples to the stage's bias voltages and supplies, introducing a multiplicative component that depends on the noise and the stage input. The model represents this effect as a multiplication in Fig. 1 of the analog residue  $r_{i-1}$  and the scaled noise  $\alpha_i n_i$ . The scale factor  $\alpha_i$  depends on processing and layout characteristics and is usually unknown. Switching noise  $n_i$  also couples to the analog input of the stage, introducing an additive component that is independent of the stage input. The model represents this effect as an addition in Fig. 1 where the scaled noise  $\beta_i n_i$  adds to the term  $r_{i-1}(1 + \alpha_i n_i)$ . Like  $\alpha_i$ , scale factor  $\beta_i$  is unknown. The result is the modified residue  $r'_{i-1}$ :

$$r'_{i-1} = r_{i-1} \underbrace{(1 + \alpha_i n_i)}_{\Delta g_i} + \underbrace{\beta_i n_i}_{\varepsilon_i}. \quad (1)$$

The term  $\alpha_i n_i$  in (1) is a *noise-induced gain error* denoted by  $\Delta g_i$  that multiplies  $r_{i-1}$ . A non-constant  $\Delta g_i$  introduces intermodulation tones in the ADC output by multiplying with the ADC input and stage residues. Because a residue contains a coarsely quantized version of the ADC input, it contains all harmonics of the input. As a result, the intermodulation tones in the ADC output appear at frequencies that are products of the harmonics and the noise. The general result of this effect has been shown in [8]. Moreover, a nonzero average of  $\Delta g_i$  scales the ADC input, causing an overall ADC gain error, while a nonzero average of  $\Delta g_i$  for  $i > 1$  introduces an interstage gain error between stages  $i-1$  and  $i$ , degrading the ADC's linearity. The term  $\beta_i n_i$  in (1) is an *additive noise error* denoted by  $\varepsilon_i$ . A nonzero  $\beta_i$  introduces an additive noise component in the ADC output that reduces the ADC's dynamic range without degrading linearity if the ADC has a sufficient correction range in every stage. Next, two switching noise cancellation algorithms are presented. One finds and cancels  $\Delta g_i$ , and the other finds and cancels  $\varepsilon_i$ .

## IV. ALGORITHMS FOR SWITCHING NOISE CANCELLATION

### A. Three-Stage Switching Noise Cancellation

Fig. 2 shows a block diagram of switching noise cancellation (SNC) at the output of the ADC. The ADC has a gain of 2 in

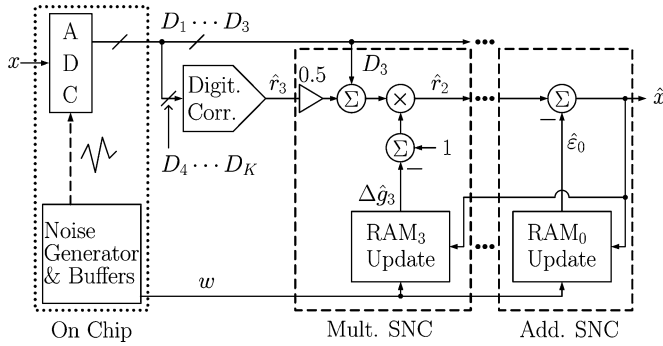


Fig. 2. 3-stage switching noise cancellation (SNC) performed off chip. Multiplicative SNC is used to cancel the noise-induced gain errors ( $\Delta g_i$ ) in the first three ADC stages ( $i = 1 \dots 3$ ). Additive SNC is used to cancel the sum of the input-referred additive noise errors ( $\varepsilon_0$ ).

each stage (i.e.,  $G = 2$  in Fig. 1), and the cancellation is done off chip using the digital outputs from each stage and the binary outputs  $w$  from the on-chip noise generator and buffers shown in the dotted box labeled as ‘On Chip’.

First, multiplicative SNC is performed in the first three stages because testing shows that SNC in these stages yields significant performance improvement. The number of stages is found empirically and was not predicted by simulation. The details for stage 3 are shown in the dashed box labeled as ‘Mult. SNC’. Let  $\hat{r}_3$  be the digitized residue output from stage 3, which is obtained after digital correction to remove redundancy from  $D_4 \dots D_K$ . Then,  $\hat{r}_3$  is scaled by 0.5 and added to  $D_3$  to obtain the digitized residue output from stage 2 before the noise error is corrected. In the analog part of the pipeline as shown in Fig. 1, for  $i = 3$ , an error was introduced by multiplying the analog residue  $r_2$  by  $1 + \Delta g_3$ , where  $\Delta g_3 = \alpha_3 n_3$ . [See (1).] This error can be corrected by dividing by  $1 + \Delta g_3$ . However, because division is difficult to implement in practice and  $\Delta g_3$  is unknown, this error is approximately corrected here by multiplying by  $1 - \Delta \hat{g}_3$ , where  $\Delta \hat{g}_3$  is an estimate of  $\Delta g_3$  if  $|\Delta g_3| \ll 1$ . The errors  $1 + \Delta g_1$  and  $1 + \Delta g_2$  in stages 1 and 2 are similarly corrected, but the block diagrams are not shown here for simplicity.

Next, additive SNC is performed. The details are shown in another dashed box labeled as ‘Add. SNC’. The additive noise terms from all the stages are canceled together through the term  $\hat{\varepsilon}_0$ , which is an estimate of the sum of the input-referred  $\varepsilon_i$  for all the stages, assuming the correction range is not exceeded in any stage. Under this condition, all interstage offsets and additive noise errors can be referred to the ADC input without introducing nonlinearity [21]. Then the new ADC output  $\hat{x}$  is used to update all the estimates of the noise-induced gain error  $\Delta \hat{g}_1 \dots \Delta \hat{g}_3$  and the estimate of the total additive noise error  $\varepsilon_0$ . The values of these estimates are stored in lookup tables implemented using RAMs as described in the next section.

The correction procedures described here do not correct for the higher-order terms that stem from interactions between  $\Delta g_i$  and  $\varepsilon_i$  in each stage (e.g.,  $\Delta g_i \varepsilon_j$  for  $i = j$  and  $i \neq j$ ). Testing reveals that these terms are below the resolution of the ADC in this work and can be ignored.

From an implementation standpoint, the delay from  $\hat{r}_3$  to  $\hat{x}$  needs to be less than one conversion cycle  $T$ , assuming the Digital Correction introduces no delay. Since three stages of SNC

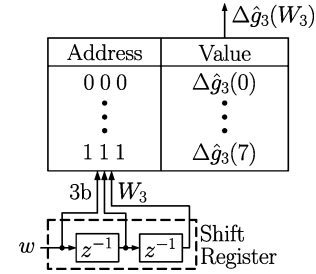


Fig. 3.  $\text{RAM}_3$  addressed by a vector formed by a shift register’s contents,  $W_3$ , which consists of the 3 most-recent outputs of the noise buffers  $w$  before the sampling instant of stage 3 in the ADC. Each period, an addressed value  $\Delta \hat{g}_3(W_3)$  is updated and used to compute a new  $\hat{x}$ .

are used in the prototype, three multipliers and four adders are present in the critical path. Ignoring the 1/2 scale operations between stages, each of these seven arithmetic functions needs to be completed within  $T/7$  or 3.6 ns for 40 MS/s conversion rate, assuming equal partitioning of the allowed delay.

### B. $\text{RAM}_3$ Lookup Table

Fig. 3 shows a diagram of a RAM used as a lookup table that stores the values of  $\Delta \hat{g}_3$ , and a shift register that addresses the RAM. The shift register input is  $w$ , which is the digital output of the noise buffers. The shift register contents form a vector of three one bit signals labeled as  $W_3$ , which addresses the RAM (i.e.,  $W_3 = \{w[m-2], w[m-1], w[m]\}$ ). The width of  $W_3$  is three bits because testing shows that the duration of the noise transient affecting a single stage is between two to three clock periods long. A longer transient requires a wider address, increasing the RAM size. When the noise buffers and the ADC are synchronous,  $W_3$  comes from the three most-recent outputs of the noise buffers before the sampling instant of stage 3, and each sample of  $W_3$  consists of 3 consecutive  $w$  values.

Each sample period, an addressed value  $\Delta \hat{g}_3(W_3)$  is read out of  $\text{RAM}_3$  to compute a new value of  $\hat{x}$ . Thus, the lookup table allows  $\Delta \hat{g}_3$  to be a general nonlinear function of  $W_3$ . One RAM lookup table corrects for the noise-induced gain error in one stage for one digital noise source. As a result, the total memory required for correction increases not only with the duration of the noise transient, but also with the number of significant and distinct digital noise sources. Since the outputs of a digital noise source (i.e.,  $w$ ) are known, noise sensors are not needed, and scaling in advanced technologies reduces the cost of integrated memory.

The use of a shift register here operating at the ADC conversion rate to detect the noise pattern is limited to noise that comes from a digital circuit operating at the same rate as the ADC. When the digital circuit operates at a faster rate than the ADC but is still synchronous with the ADC’s sample rate, the shift register needs to operate at the faster rate to capture the entire noise pattern. Then the shift register output can be down sampled to the ADC’s output rate so that the lookup table is updated at the same rate as the ADC output. If the noise comes from an asynchronous source, the noise pattern detected by the shift register does not provide enough information to determine the effect of the noise because asynchronous noise can occur at any time during the conversion period.



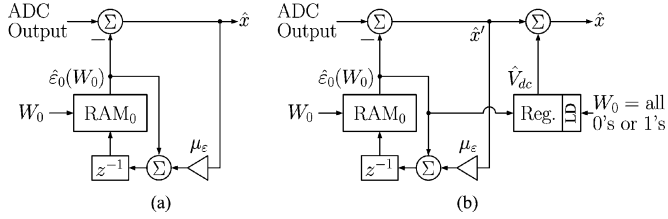


Fig. 6. Block diagram for updating  $\hat{\epsilon}_0$  (the estimate of the sum of the input-referred  $\epsilon_i$ ) (a) with and (b) without a DC null in  $\hat{x}$ . In (b), a register is used to store  $\hat{V}_{dc}$  (the estimate of the DC component at the ADC input). The register only loads (LD)  $\hat{V}_{dc}$  when  $W_0$  contains either all 0's or 1's.

Because  $\text{NFB}_2$  drives  $\bar{e} \rightarrow 0$  (Section IV-D),  $\overline{s^2/(1+\hat{\rho})}/\sigma^2 \rightarrow 1$ , which can be approximated<sup>1</sup> by  $(s^2 - s^2\hat{\rho})/\sigma^2 \rightarrow \sigma^2$  if  $|\hat{\rho}| \ll 1$ . Because  $\overline{s^2} = \sigma^2$  by definition and  $\overline{s^2\hat{\rho}} = \overline{s^2} \cdot \overline{\hat{\rho}}$  under the assumption that  $s$  and  $\hat{\rho}$  are independent,  $\overline{\hat{\rho}} \rightarrow 0$  under the condition  $\hat{s} = s$ . Then, because  $\text{NFB}_1$  and  $\text{NFB}_2$  are coupled, reducing the error  $\hat{\rho}$  in  $\hat{\sigma}^2$  reduces the error in  $\Delta\hat{g}$ .

In summary, correct convergence requires three conditions: 1) The input and the noise are statistically independent. 2) The RMS noise is smaller than the RMS input. 3) The RAM address  $W$  sometimes contains either all 0's or all 1's (i.e.,  $\hat{s} = s$ ).

#### F. To Update $\epsilon_0$

Fig. 6(a) shows the adaption process for finding the estimate of the sum of the input-referred additive noise errors,  $\hat{\epsilon}_0$ . The subscript 0 indicates the error terms are summed and input referred. Testing reveals that cancellation in the first three ADC stages is sufficient and more stages yields insignificant additional improvement. Similar to  $W_3$  in Section IV-B,  $W_0$  is a vector that contains the four most-recent outputs of the noise buffers before the sampling instant of stage 3. Four bits are needed here because the total duration of the noise transients affecting the first three stages together is one clock period longer than for a single stage.

The estimate  $\hat{\epsilon}_0$  is found by a RAM-based LMS adaptation loop such as the one in [25], which can be viewed as an implementation and extension of the general concept of noise cancellation in [26]. The loop begins with the corrected ADC output  $\hat{x}$  and updates  $\hat{\epsilon}_0$  in  $\text{RAM}_0$  at a location addressed by  $W_0$  each sample period. When  $\hat{x}(W_0) \rightarrow 0$ ,  $\hat{\epsilon}_0$  overcomes the effect of  $\epsilon_0$ , and  $\hat{\epsilon}_0(W_0) \approx \epsilon_0(W_0)$  if the higher-order terms described at the end of Section IV-A are ignored.

If the ADC input  $x$  contains a nonzero DC component, all values in the RAM contain this component. Also, the ADC's input-referred offset  $V_{os}$  adds to the input at DC. Let  $V_{dc}$  denote the sum of the nonzero DC component in  $x$  and  $V_{os}$ . Because  $V_{dc}$  is in all RAM values and these values are subtracted from the ADC output, the algorithm introduces a DC null in the frequency spectrum of  $\hat{x}$ . To overcome this problem, a modified algorithm is used.

Fig. 6(b) shows the modification where  $\hat{x}'$  denotes the ADC output with a DC null. Also, the modification includes a new register and an extra summer. When  $W_0$  contains either all 0's

<sup>1</sup>The approximation here is done by using Taylor series expansion that  $1/(1+\hat{\rho}) \approx 1-\hat{\rho}$  if  $|\hat{\rho}| \ll 1$ . Because Taylor series expansion is valid on every sample of  $s$  and  $\hat{\rho}$ , it is also valid on the ensembles of  $s$  and  $\hat{\rho}$ .

TABLE II  
ESTIMATED DIE AREA AND POWER DISSIPATION OF THE SNC BLOCKS IN A 0.18  $\mu\text{m}$  CMOS TECHNOLOGY

$f_S = 40 \text{ MS/s}$	Area ( $\text{mm}^2$ )	Power (mW)
Multiplicative SNC per stage	0.08	1.67
Additive SNC	0.09	0.39
Total for 3 stages	0.33	5.40

or all 1's, no digital noise occurs. Thus in these cases, the ADC output contains  $x(W_0) + V_{os}$  without noise, and the algorithm estimates  $V_{dc}$  by finding  $\hat{\epsilon}(W_0)$  after convergence. Let  $\hat{V}_{dc}$  denote the estimate of  $V_{dc}$ . A new value of  $\hat{V}_{dc}$  is loaded into the new register in Fig. 6(b) only during the noiseless cases. The value of  $\hat{V}_{dc}$  is read out of the register and added to every sample of  $\hat{x}'$  to produce a new  $\hat{x}$  that does not have a DC null. This technique removes the additive noise errors introduced by switching noise coupling, but it does not remove the ADC offset. To remove the offset, other calibration methods such as the one in [27] can be used.

#### G. Area and Power Dissipation Estimates

In practice, division in (3) can be avoided for simplicity by using  $e'$  rather than  $e$ , where

$$e' = \hat{\sigma}^2 e = \mu_g (\hat{s}^2 - \hat{\sigma}^2). \quad (5)$$

This simplification makes the adaptation process in Section IV-D converge more slowly for the same step size  $\mu_g$  because  $e'$  in (5) is smaller than  $e$  in (3) for  $\hat{\sigma}^2 < 1$ . However, implementation with (5) avoids a numerical problem in (3) when  $\hat{\sigma}^2 = 0$ , and increasing  $\mu_g$  introduces more steady-state variation in  $e'$ . Starting with a large value of  $\mu_g$  and reducing  $\mu_g$  as the adaptation approaches convergence gives both fast convergence and small steady-state variation but was not done in the prototype. With this simplified implementation, Table II shows the estimated die area and power dissipation of the proposed 3-stage SNC blocks if implemented in a 0.18  $\mu\text{m}$  CMOS technology at a conversion rate  $f_S = 40 \text{ MHz}$ .

## V. PROTOTYPE

Fig. 7(a) shows a block diagram of the floor plan of the prototype chip. The pipelined ADC's outputs are driven off chip through small drivers at the bottom of the figure. A bank of eight large and identical noise buffers whose gates are connected together is included at the top of the figure. A noise buffer is a digital output driver that consists of four tapered inverters. Furthermore, to ensure the noise buffers are the dominant source of switching noise, the ADC output is down-sampled on chip by a factor of six, reducing the required number of ADC drivers. Finally, by connecting the power supply of the noise buffers ( $V_{ddn}$ ) to the contacts of an N-well adjacent to the first pipeline stage, significant switching noise couples to the ADC via the substrate.

To demonstrate SNC, all the noise buffers are driven by the same input that is synchronous to the ADC through a multiplexer (MUX) that selects one of four noise sources: an external

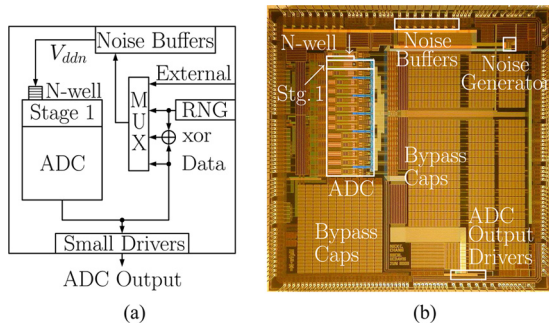


Fig. 7. Prototype chip's (a) floor plan and (b) die photograph. A noise buffer is a large digital output buffer. By connecting the power supply of the noise buffers ( $V_{ddn}$ ) to the contacts of an N-well adjacent to the first pipeline stage, significant switching noise couples to the ADC via the substrate.

source (External), a random number generator output (RNG), an ADC output without scrambling (Data), or the same ADC output with scrambling (Data xor RNG) [28]. The ADC output used for noise coupling is a raw digital output from the first stage in a 1.5 bit per stage pipeline. This binary output is 1 when the ADC input is greater than the comparator threshold  $V_{ref}/4$  and 0 otherwise, where  $\pm V_{ref}$  denotes the input full-scale voltage of the ADC. Scrambling is done by 'XORing' this ADC output bit with a random binary signal. Finally, the external source generates a square wave to provide a periodic noise pattern. Fig. 7(b) shows the corresponding die photograph of the prototype chip, which was fabricated in a  $0.18 \mu\text{m}$  CMOS process on a lightly-doped substrate. The area of the chip is  $25 \text{ mm}^2$ . The pipelined ADC consists of 12 1.5 bit stages followed by a 2 bit flash converter. The active area of the ADC is  $3 \text{ mm}^2$ . The bank of noise buffers is on the top side of the chip. The locations of the first pipeline stage and the noise-injected N-well are also shown.

## VI. MEASURED RESULTS

All measured results come from the simplified implementation described in Section IV-G. The conversion rate of the prototype is limited by the settling time of the interstage amplifiers in the pipelined ADC and is set to 40 MS/s in the tests below except where stated otherwise.

### A. Periodic Noise

Fig. 8 shows three integral-nonlinearity plots: without switching noise (top), with periodic switching noise injected into the N-well before SNC (middle), and with this noise after SNC (bottom). In this paper, all the code density tests [29] have a resolution of 12 bits and use an input amplitude that covers about 90% of the full-scale range. The other 10% of the range is saved to accommodate the switching noise without exceeding the full-scale range. A similar approach has been used previously when dither has been added to the input [30]. Without switching noise, the noise buffers are inactive, and the maximum magnitude of the INL in the top plot of Fig. 8 is about 0.7 LSB. With periodic noise, the maximum INL in the middle plot increases to 2.3 LSB. The main observation here is that the INL pattern resembles the pattern caused by interstage gain errors. This result stems from the fact that noise-induced

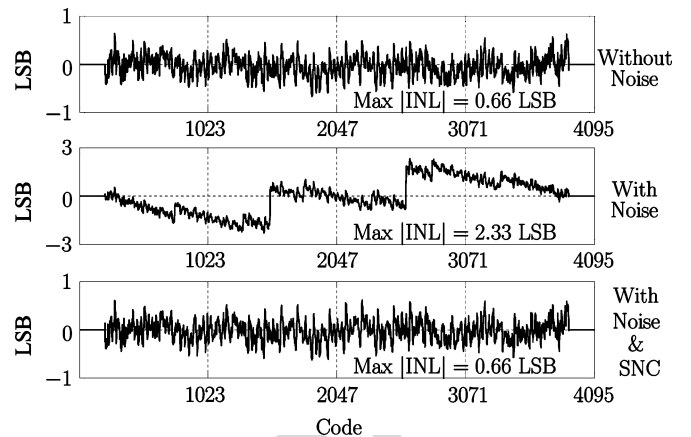


Fig. 8. INL plots related to periodic noise.

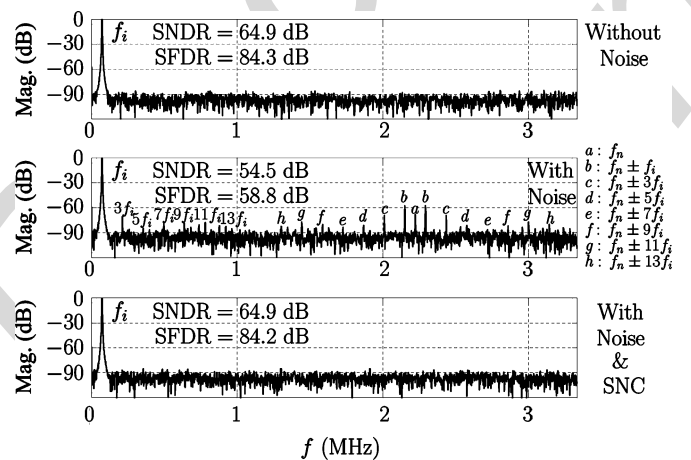


Fig. 9. FFT plots related to periodic noise.

gain error changes the effective gain applied to the stage input. Also, the middle INL plot has a downward trend in the regions from code 0 to 1500, code 1500 to 2500, and code 2500 to 4000 with vertical jumps upward at codes 1508 and 2591. This downward trend shows the effective interstage gain between the first and second stages is greater than the ideal value of 2 due to the switching noise. This effective gain error corresponds to the average value of  $\Delta g_2$  because  $\Delta g_2$  scales the analog residue output from stage 1 (and  $\Delta g_1$  scales the ADC input), as described in Section III. After SNC, the maximum INL in the bottom plot of Fig. 8 is about the same as in the case without noise.

Fig. 9 shows three FFT plots: without switching noise, with periodic switching noise before SNC, and with this noise after SNC. In all three cases, the conversion rate  $f_s = 40 \text{ MS/s}$ , and the input frequency  $f_i = 71 \text{ kHz}$ . Also, because the output is down-sampled on chip by a factor of 6, the x-axes on the plots extend to half the down-sampled conversion rate,  $1/2 \times f_s/6 \approx 3.3 \text{ MHz}$ . Since the ADC output is down-sampled in this project, the noise cancellation (i.e., multiplication and subtraction to compute  $\hat{x}$  in Fig. 2) is performed at the down-sampled rate, and the RAM updates are also performed at that rate. Without switching noise as shown in the top plot of Fig. 9, the SNDR and SFDR are 64.9 dB and 84.3 dB, respectively. With

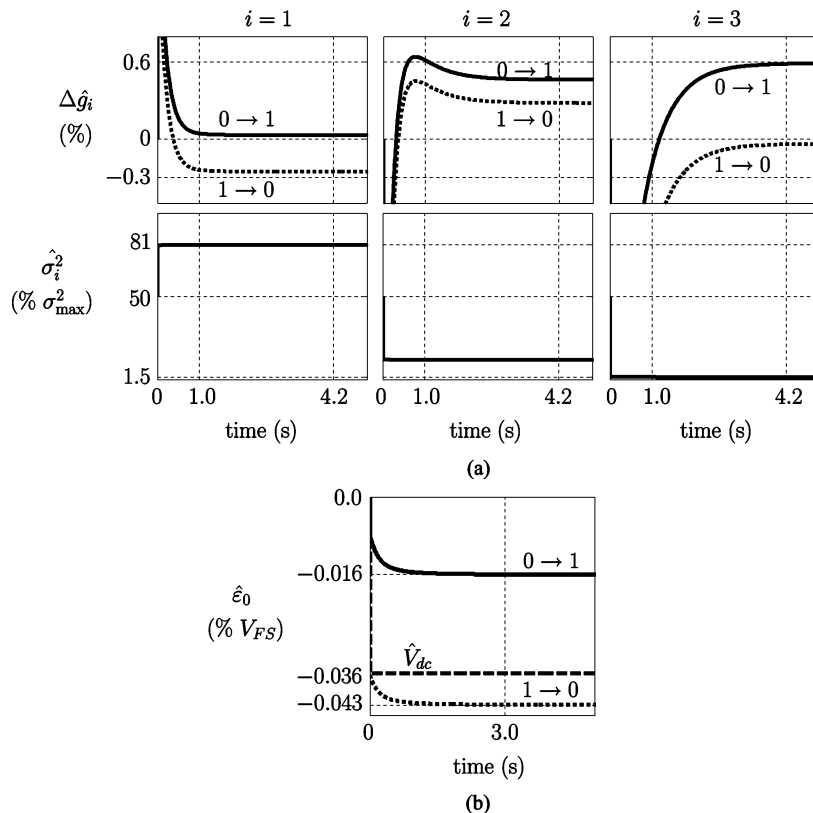


Fig. 10. Measured plots for (a)  $\Delta\hat{g}_i$  and  $\hat{\sigma}_i^2$ , and (b)  $\hat{\varepsilon}_0$  versus time for the periodic-noise case. The plots labeled as  $0 \rightarrow 1$  and  $1 \rightarrow 0$  show values of the noise estimates that correspond to switching the noise buffers from 0 to 1 and 1 to 0, respectively, right before the sampling instant of the first ADC stage.

all eight noise buffers turned on, switching noise appears and the SNDR and SFDR degrade to 54.5 dB and 58.8 dB, respectively. The switching noise here is generated by switching the noise buffers with a square wave at a frequency  $f_n$ , which is 1/3 of the down-sampled conversion rate,  $(1/3) \times (f_s/6) \approx 2.2$  MHz. Although higher-order noise tones may be hidden in the FFT plot by the choice of  $f_n$ , the chosen frequency allows for large noise coupling to the ADC after down-sampling. Because  $f_n$  is 1/3 of the down-sampled conversion rate, 2/3 of the down-sampled ADC outputs are affected by  $0 \rightarrow 1$  and  $1 \rightarrow 0$  transitions of the square-wave noise. The remaining 1/3 of the ADC outputs correspond to the noiseless samples required by the convergence condition specified in Section IV-E. With this noise, the main difference between the two top plots appears at frequencies labeled as  $a, b, \dots, h$ . The tone at  $a$  stems from additive noise and appears at  $f_n = 2.2$  MHz. The tones at  $b, c, \dots, h$  stem from intermodulation between the harmonics of the input at  $f_i$  and the switching noise at  $f_n$  (e.g.,  $f_n \pm f_i$  and  $f_n \pm 3f_i$ ). Also with noise, input harmonic tones (e.g.,  $3f_i$ ) appear because switching noise causes interstage gain errors as explained with the INL plots in Fig. 8. After SNC, the intermodulation tones and other undesired noise components are suppressed as shown in the bottom plot of Fig. 9. The SNDR and SFDR in the bottom case become 64.9 dB and 84.2 dB, respectively. SNC suppresses periodic noise in the prototype generated by noise buffers operating synchronously with the ADC for all frequencies up to half the sampling rate under the assumptions stated in Section IV-E.

The top of Fig. 10(a) shows measured plots for the estimates of the noise-induced gain errors in the first three stages,  $\Delta\hat{g}_i$ ,

versus time, where  $i$  indicates stage 1, 2, and 3. The values of  $\Delta\hat{g}_i$  are plotted in % and range from  $-0.3\%$  to  $+0.6\%$  after convergence. For each stage, the solid curve corresponds to switching the noise buffers from  $0 \rightarrow 1$  right before the sampling instant of the first ADC stage, and the dotted curve corresponds to  $1 \rightarrow 0$  transitions at the same instant.

The bottom of Fig. 10(a) shows the corresponding plots for the estimates of the mean-square values of the digitized stage inputs,  $\hat{\sigma}_i^2$  for  $i = 1, 2, 3$ , versus time. The values of these estimates are plotted in % of  $\sigma_{\max}^2$ , which is the mean-square value of a sinusoidal input that covers the full-scale input range. The values of these estimates range from 1.5% to 81% after convergence.

In general, the convergence time of  $\Delta\hat{g}_i$  depends mainly on the occurrence of noise samples at the ADC output; convergence is faster when noise samples occur more frequently. Also,  $\hat{\sigma}_i^2$  converges faster than  $\Delta\hat{g}_i$  because  $\hat{\sigma}_i^2$  is always updated with every sample of  $e$  in (3), and  $\Delta\hat{g}_i$  is only updated when a noise transition occurs.

Fig. 10(b) shows three measured plots for the estimate of the input-referred additive noise error  $\hat{\varepsilon}_0$  versus time. The first two plots are for  $\hat{\varepsilon}_0$ , where the solid curve is for  $0 \rightarrow 1$  transitions right before the sampling instant of the first ADC stage, and the dotted curve is for  $1 \rightarrow 0$  transitions at the same instant. Both plots include the estimated DC component  $\hat{V}_{dc}$  at the ADC input (about  $-0.036\%$  of full scale). The third plot is dashed and shows  $\hat{V}_{dc}$ . The values of  $\hat{\varepsilon}_0$  are plotted in % of full scale and range from  $-0.016\%$  to about  $-0.043\%$  after convergence.



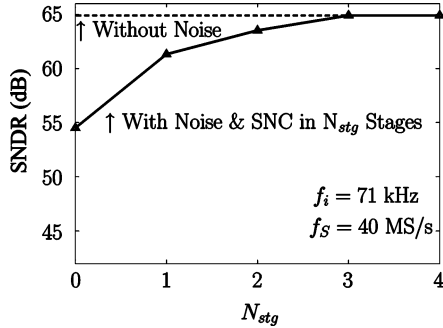


Fig. 11. SNDR versus the number of stages  $N_{\text{stg}}$  in which SNC is applied, for the periodic-noise case.

This estimate converges within 3 seconds for  $f_S = 40$  MS/s and for a step size  $\mu_\varepsilon = 2^{-14}$ .

In the periodic-noise case, the effective noise-induced gain error  $\Delta g_2$  can be calculated from Fig. 10(a). Both  $0 \rightarrow 1$  and  $1 \rightarrow 0$  transitions affect the ADC output once every three samples because  $f_n$  is  $1/3$  of the down-sampled conversion rate. From Fig. 10(a), the estimated values of  $\Delta g_2$  for  $0 \rightarrow 1$  and  $1 \rightarrow 0$  transitions are about 0.45% and 0.30%, respectively. Thus their average value is about  $(0.45\% + 0.30\%) / 3 = 0.25\%$ , which is about the same as the calculated  $\Delta g_2$  inferred from the measured INL plot ( $\sim 0.24\%$ ).

Fig. 11 plots the SNDR versus the number of stages  $N_{\text{stg}}$  with SNC. When  $N_{\text{stg}} = 1$ , only the first stage is corrected. When  $N_{\text{stg}} = 3$ , the first three stages are all corrected, and the SNDR is the same as without noise; therefore, no further correction is needed. An overall improvement of 10 dB is seen. About 7 dB of this improvement comes from correcting just the first stage.

### B. Random Noise

Fig. 12 shows three FFT plots: without deliberately injected switching noise, with switching noise that comes from the random number generator before SNC, and with this random noise after SNC. Again,  $f_i = 71$  kHz. Without switching noise, the noise buffers are turned off and the SNDR and SFDR are 64.9 dB and 84.3 dB, respectively. With all eight noise buffers driven by the same random noise, the noise floor is raised, degrading the SNDR to 51.9 dB (3 dB worse than with periodic noise). Also, the SFDR falls to 75.3 dB, which is about 17 dB higher than with periodic noise because the random noise is white, unlike the periodic noise. Since the noise is white, every switching pattern,  $W = 000, \dots, 111$ , is equally likely. So, each pattern occurs one in eight samples on average, and a noiseless pattern ( $W = 000$  or  $W = 111$ ) occurs one in four samples on average. Finally, SNC reduces the noise floor and restores the SNDR and SFDR to nearly their levels without switching noise.

Fig. 13 shows plots of SNDR and SFDR versus  $f_S$  without and with periodic and random noise for conversion rates from 5 MS/s to 40 MS/s. Each case shows three curves. The solid curve represents the case without noise. The dashed curve represents the case with noise but without SNC. The dotted curve just below the solid curve represents the case with noise and with SNC. The ADC input is sinusoidal at  $f_i = 71$  kHz in all cases. Turning on the noise without SNC introduces a smaller

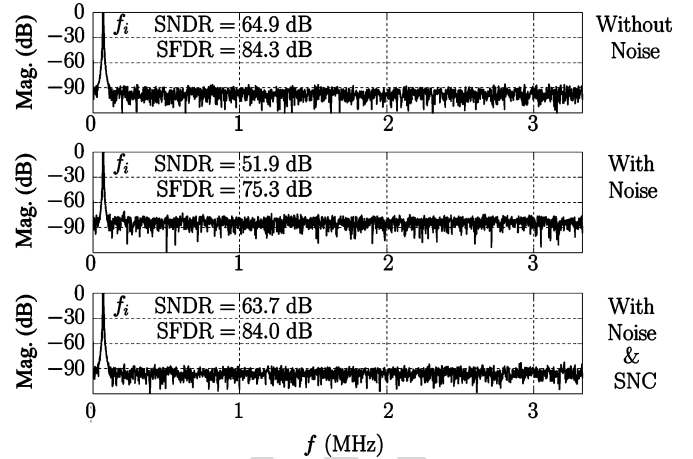


Fig. 12. FFT plots related to random noise.

penalty at 5 MS/s than at 40 MS/s. For example, in Fig. 13(a), the lower, dashed plot with noise but without SNC shows the SNDR at 5 MS/s is about 13 dB higher than at 40 MS/s. Also, in Fig. 13(b), the SFDR at 5 MS/s is about 15 dB higher than at 40 MS/s. The smaller penalties stem from the fact that the transients introduced by the noise have enough time to decay to insignificant levels between samples at 5 MS/s. Furthermore, in Figs. 13(a) and (b), the SNDR and SFDR at 20 MS/s in the dashed plot are about 3 dB worse than at 40 MS/s. This higher penalty at a lower frequency shows that the noise transients do not always add constructively with increasing frequency. Finally, for the periodic-noise case, turning on SNC just about restores the performance to the original levels. For the random-noise case, turning on SNC restores the performance within 2 dB of the original levels.

The periodic-noise case converges faster than the random-noise case (e.g.,  $\sim 120$  k samples versus  $\sim 1$  M samples) because periodic noise always drives the error function (3) in the same direction until convergence is achieved. In the random-noise case, convergence requires more samples to average out the random fluctuation in the error function, unlike the periodic-noise case.

### C. ADC Output Noise With Scrambling

For this case, the noise is caused by an output bit from the first ADC stage with scrambling that drives all 8 noise buffers. Scrambling is required because the noise is otherwise input dependent, and the SNC algorithm is not effective in that case. The noise without and with scrambling will be referred to as *unscrambled* and *scrambled noise*, respectively. The scrambled noise does not generate harmonic tones, but the noise floor in the FFT plot (not shown for brevity) rises. This effect stems from a whitening of the noise floor provided by scrambling, similar to the random-noise case shown in Section VI-B. Also, because the ADC output bit that drives the noise buffers changes so slowly that insignificant noise is generated with  $f_i = 71$  kHz,  $f_i$  is increased to 9.112 MHz here. This input frequency is about 100 times higher than in the previous cases and is chosen so that the performance degradation with noise and without SNC is similar to that in the periodic-noise case.

TABLE III  
 PERFORMANCE SUMMARY

Resolution	12 bits										
Conversion Rate $f_S$	40 MS/s										
Technology	0.18- $\mu\text{m}$ CMOS, Lightly-doped										
Supply Voltages	1.8 V (ADC and Noise Generator), 3.3 V (Noise Buffers and ADC Output Buffers)										
Full-Scale $V_{FS}$	1.6 V <sub>p-p</sub> Differential										
Active Area	3 mm <sup>2</sup> (ADC), 1 mm <sup>2</sup> (Noise Generator & Buffers)										
Power	99 mW (Analog), 17 mW (Digital)										
Input Frequency $f_i$	71 kHz					9.112 MHz					
Noise Type	None	Periodic		Random		None	Unscrambled <sup>+</sup>		Scrambled <sup>+</sup>		
SNC	No	No	Yes	No	Yes	No	No	Yes	No	Yes	
INL [LSB]	0.66	2.33	0.66	1.74	0.69	0.66	1.72	n/a	1.64	0.78	
DNL [LSB]	0.26	0.67	0.26	0.50	0.31	0.24	0.45	n/a	0.45	0.25	
SNDR [dB]	64.9	54.5	64.9	51.9	63.7	63.4	55.8	n/a	51.8	62.1	
SFDR [dB]	84.3	58.8	84.2	75.3	84.0	83.3	59.7	n/a	70.6	83.1	
ENOB [bits]	10.5	8.8	10.5	8.3	10.3	10.2	9.0	n/a	8.3	10.0	

+ ADC output without or with scrambling.

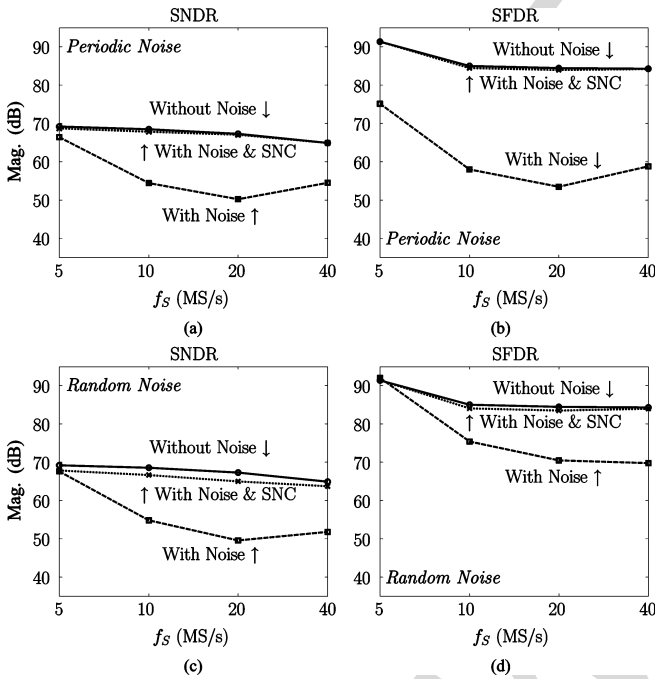


Fig. 13. SNDR and SFDR versus conversion rate  $f_S$  for two noise cases: (a)-(b) periodic noise and (c)-(d) random noise.  $f_i = 71$  kHz.

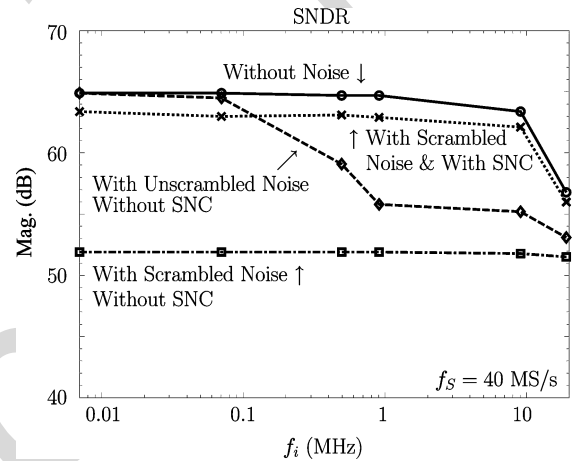


Fig. 14. SNDR versus input frequency  $f_i$  for the ADC output noise without and with scrambling.  $f_S = 40$  MHz.

Fig. 14 shows four plots of SNDR versus  $f_i$  in log scale for  $f_S = 40$  MS/s. The x-axis extends to the Nyquist frequency, 20 MHz. The solid curve is without noise. The SNDR in that case drops after about 10 MHz because of jitter on the

sample clock. The other three cases are with noise. The dashed curve is with unscrambled noise and without SNC. The SNDR falls monotonically with increasing  $f_i$  in this case. This trend stems from the fact that increasing  $f_i$  causes the ADC outputs to change more frequently and introduce more noise, degrading the SNDR. Since SNC is not helpful without scrambling, the case with unscrambled noise and with SNC is omitted in Fig. 14. The last two cases are with scrambled noise. The case without SNC is at the bottom of Fig. 14 in the dash-dot curve, and the case with SNC is second from the top in the dotted curve. With scrambled noise and without SNC, the level of noise does not

depend on the input frequency  $f_i$ . In this case, the SNDR is about the same as with random noise, which is about 11 dB down from the case without noise. With scrambled noise and with SNC, the SNDR is within 2 dB of the case without noise.

## VII. CONCLUSION

Table III summarizes performance. This work shows that the additive and multiplicative noise errors caused by switching noise in a pipelined ADC can be found and canceled at the output using a digital background adaptive technique. The proposed technique does not require noise sensors, and this technique is mainly useful when the noise is dominated by the activity of a small number of digital signals. In practice, this technique may be combined with the traditional strategies such as guard banding and physical separation to minimize the cost of noise reduction in area and power dissipation.

## REFERENCES

- [1] K. Bult, "Broadband communication circuits in pure digital deep sub-micron CMOS," in *IEEE Int. Solid-State Circuits (ISSCC) Dig. Tech. Papers*, 1999, pp. 76–77.
- [2] M. Badaroglu *et al.*, "Modeling and experimental verification of substrate noise generation in a 220-Kgates WLAN system on-chip with multiple supplies," *IEEE J. Solid-State Circuits*, vol. 38, no. 7, pp. 1250–1260, Jul. 2003.
- [3] K.-S. Ha *et al.*, "A 0.13- $\mu$ m CMOS 6 Gb/s/pin memory transceiver using pseudo-differential signaling for removing common-mode noise due to SSN," *IEEE J. Solid-State Circuits*, vol. 44, no. 11, pp. 3146–3162, Nov. 2009.
- [4] B. Brandt and B. Wooley, "A 50-MHz multibit Sigma-Delta modulator for 12-b 2-MHz A/D conversion," *IEEE J. Solid-State Circuits*, vol. 26, no. 12, pp. 1746–1756, Dec. 1991.
- [5] M. Ito *et al.*, "A 10 bit 20 MS/s 3 V supply CMOS A/D converter," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1531–1536, Dec. 1994.
- [6] T. Blalack and B. Wooley, "The effects of switching noise on an over-sampling A/D converter," in *IEEE Int. Solid-State Circuits (ISSCC) Dig. Tech. Papers*, 1995, pp. 200–201, 367.
- [7] M. Mayes and S. W. Chin, "A 200 mW, 1 MS/s, 16-b pipelined A/D converter with on-chip 32-b microcontroller," *IEEE J. Solid-State Circuits*, vol. 31, no. 12, pp. 1862–1872, Dec. 1996.
- [8] M. Xu *et al.*, "Measuring and modeling the effects of substrate noise on the LNA for a CMOS GPS receiver," *IEEE J. Solid-State Circuits*, vol. 36, no. 3, pp. 473–485, Mar. 2001.
- [9] X. Aragonés *et al.*, *Analysis and Solutions for Switching Noise Coupling in Mixed-Signal ICs*, 1st ed. Delft, The Netherlands: Kluwer Academic, 1999.
- [10] M. Peng and H.-S. Lee, "Study of substrate noise and techniques for minimization," *IEEE J. Solid-State Circuits*, vol. 39, no. 11, pp. 2080–2086, Nov. 2004.
- [11] P. van Zeijl *et al.*, "A Bluetooth radio in 0.18- $\mu$ m CMOS," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1679–1687, Dec. 2002.
- [12] C. Soens *et al.*, "Modeling of substrate noise generation, isolation, and impact for an LC-VCO and a digital modem on a lightly-doped substrate," *IEEE J. Solid-State Circuits*, vol. 41, no. 9, pp. 2040–2051, Sep. 2006.
- [13] W.-K. Yeh *et al.*, "Substrate noise-coupling characterization and efficient suppression in CMOS technology," *IEEE Trans. Electron Devices*, vol. 51, no. 5, pp. 817–819, May 2004.
- [14] A. Ali *et al.*, "A 14 bit 125 MS/s IF/RF sampling pipelined ADC with 100 dB SFDR and 50 fs jitter," *IEEE J. Solid-State Circuits*, vol. 41, no. 8, pp. 1846–1855, Aug. 2006.
- [15] R. Rossi *et al.*, "Model and verification of triple-well shielding on substrate noise in mixed-signal CMOS ICs," in *Proc. 29th ESSCIRC*, 2003, pp. 643–646.
- [16] K. Makie-Fukuda *et al.*, "Substrate noise reduction using active guard band filters in mixed-signal integrated circuits," in *Symp. VLSI Circuits Dig. Tech. Papers*, 1995, pp. 33–34.

- [17] T. Nakura *et al.*, "Feedforward active substrate noise canceling technique using power supply di/dt detector," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2005, pp. 284–287.
- [18] M. Nagata *et al.*, "Effects of power-supply parasitic components on substrate noise generation in large-scale digital circuits," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2001, pp. 159–162.
- [19] M. van Heijningen *et al.*, "Analysis and experimental verification of digital substrate noise generation for epi-type substrates," *IEEE J. Solid-State Circuits*, vol. 35, no. 7, pp. 1002–1008, Jul. 2000.
- [20] G. Van der Plas *et al.*, "High-level simulation of substrate noise in high-ohmic substrates with interconnect and supply effects," in *Proc. 41st Design Automation Conf.*, 2004, pp. 854–859.
- [21] S. Lewis and P. Gray, "A pipelined 5-MS/s 9 bit analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. SSC-22, no. 6, pp. 954–961, Dec. 1987.
- [22] A. J. Bell and T. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.
- [23] D. J. MacKay, "Maximum Likelihood and Covariant Algorithms for Independent Component Analysis," Univ. Cambridge, Cambridge, U.K., Tech. Rep., 1996.
- [24] J.-F. Cardoso, "Infomax and maximum likelihood for blind source separation," *IEEE Signal Process. Lett.*, vol. 4, no. 4, pp. 112–114, Apr. 1997.
- [25] K. Fisher *et al.*, "An adaptive RAM-DFE for storage channels," *IEEE Trans. Commun.*, vol. 39, no. 11, pp. 1559–1568, Nov. 1991.
- [26] B. Widrow *et al.*, "Adaptive noise canceling: Principles and applications," *Proc. IEEE*, vol. 63, no. 12, pp. 1692–1716, Dec. 1975.
- [27] S. Jamal *et al.*, "A 10-b 120-MS/s time-interleaved analog-to-digital converter with digital background calibration," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1618–1627, Dec. 2002.
- [28] R. Jewett *et al.*, "A 12b 128 MS/s ADC with 0.05 LSB DNL," in *IEEE Int. Solid-State Circuits (ISSCC) Dig. Tech. Papers*, 1997, pp. 138–139, 443.
- [29] J. Doernberg *et al.*, "Full-speed testing of A/D converters," *IEEE J. Solid-State Circuits*, vol. SSC-19, no. 6, pp. 820–827, Dec. 1984.
- [30] J. Ming and S. Lewis, "An 8 bit 80-MS/s pipelined analog-to-digital converter with background calibration," *IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1489–1497, Oct. 2001.



**Nick C.-J. Chang** received the B.S. and M.S. degrees from Columbia University, New York, NY, USA, and the Ph.D. degree from the University of California, Davis, CA, USA, all in electrical engineering.

In 2011, he joined Marvell Semiconductor, where he built an ADC for an Ethernet transceiver. His research interests include data conversion technique, digital signal processing, and analog circuit design.



**Paul J. Hurst** (S'76–M'83–SM'94–F'01) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California, Berkeley, CA, USA, in 1977, 1979, and 1983, respectively.

From 1983 to 1984, he was with the University of California, Berkeley, as a lecturer, teaching integrated-circuit design courses and working on an MOS delta-sigma modulator. In 1984, he joined Silicon Systems Inc., Nevada City, CA, USA, where he was involved in the design of CMOS integrated circuits for voice-band modems. Since 1986, he has

been on the faculty of the Department of Electrical and Computer Engineering at the University of California at Davis, where he is now a Professor. His research interests are in the areas of data converters and analog and mixed-signal integrated-circuit design for digital communications. He is a co-author of a textbook on analog integrated-circuit design. He is also active as a consultant to industry.

Prof. Hurst has served on the program committees for the Symposium on VLSI Circuits and the IEEE International Solid-State Circuits Conference. He has served as an associate editor for the IEEE JOURNAL OF SOLID-STATE CIRCUITS and as a member of the administrative committee of the IEEE Solid-State Circuits Society.



**Bernard C. Levy** received the diploma of Ingénieur Civil des Mines from the Ecole Nationale Supérieure des Mines in Paris, France, in 1974, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 1979.

From July 1979 to June 1987, he was an Assistant and then Associate Professor in the Department of Electrical Engineering and Computer Science at Massachusetts Institute of Technology (M.I.T.), Cambridge, MA, USA. Since July 1987, he has been with the University of California, Davis, CA, USA,

where he is Professor of Electrical Engineering and a member of the Graduate Group in Applied Mathematics. He served as Chair of the Department of Electrical and Computer Engineering at UC Davis from 1996 to 2000. He was a Visiting Scientist at the Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA) in Rennes, France from January to July 1993, and at the Institut National de Recherche en Informatique et Automatique (INRIA), in Rocquencourt, France, from September to December 2001. He is the author of the book *Principles of Signal Detection and Parameter Estimation* (Springer, 2008). His research interests include statistical signal processing, estimation, detection, and circuits applications of signal processing.

Dr. Levy served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I, the IEEE TRANSACTIONS ON CIRCUITS AND

SYSTEMS II, and the EURASIP *Journal on Advances in Signal Processing*. He is currently an Associate Editor of *Signal Processing*. He is a member of SIAM.



**Stephen H. Lewis** (S'85–M'88–SM'97–F'01) received the B.S. degree from Rutgers University, New Brunswick, NJ, USA, in 1979, the M.S. degree from Stanford University, Stanford, CA, USA, in 1980, and the Ph.D. degree from the University of California, Berkeley, CA, USA, in 1987, all in electrical engineering.

From 1980 to 1982, he was with Bell Laboratories, Whippany, NJ, USA, where he was involved in circuit design for magnetic recording. In 1988, he re-joined Bell Laboratories, Reading, PA, USA, where he concentrated on the design of analog-to-digital converters. In 1991, he joined the Department of Electrical and Computer Engineering, University of California, Davis, CA, USA, where he is now a Professor. He is a co-author of a college textbook on analog integrated circuits, and his research interests include data conversion, signal processing, and analog circuit design.

IEEE Pre-proof  
Web Version

# Background Adaptive Cancellation of Digital Switching Noise in a Pipelined Analog-to-Digital Converter Without Noise Sensors

Nick C.-J. Chang, Paul J. Hurst, *Fellow, IEEE*, Bernard C. Levy, and Stephen H. Lewis, *Fellow, IEEE*

**Abstract**—Switching noise generated by digital circuits can degrade analog circuit performance in mixed-signal integrated circuits (ICs). In an analog-to-digital converter (ADC), one major source of switching noise is digital output drivers. Traditional methods for mitigating this problem mostly have been to isolate the analog and digital circuits to minimize digital noise coupling into sensitive analog nodes. This paper presents two fully digital and adaptive algorithms, which find and cancel errors due to switching noise coupling at the output of an ADC without using noise sensors. To demonstrate the operation of these algorithms, a 12 bit, 40 MS/s pipelined ADC has been designed and fabricated in 0.18  $\mu\text{m}$  CMOS process. The system consists of an ADC with its own output drivers, and eight other independent digital output drivers (noise buffers) that can be programmed to produce four different kinds of switching noise. The switching noise cancellation (SNC) algorithm estimates noise parameters and stores them in look-up tables. At the ADC output, the effects of switching noise are digitally removed to recover the input samples. Test results show that the ADC achieves a signal-to-noise-and-distortion-ratio (SNDR) of 64.9 dB with all of the noise-generating buffers off. With the noise buffers on, the worst case SNDR before and after SNC is 51.9 dB and 63.7 dB, respectively.

**Index Terms**—additive noise, analog-to-digital converter, background adaptive noise cancellation, digital switching noise, least-mean-square estimation, LMS, maximum-likelihood estimation, multiplicative noise, noise-induced gain error pipelined ADC, power supply bounce, simultaneous switching noise, SSN, substrate noise.

## I. INTRODUCTION

**A**NALOG and digital circuits are routinely included together on integrated circuits, and the trend is toward full system integration on a single chip. The main potential advantage of this approach is that it can reduce cost and power dissipation. Unfortunately, integrating analog and digital circuits on

the same die degrades the raw performance of analog circuits because of switching noise coupling from the digital side [1]. In many mixed-signal IC designs, digital output drivers are a dominant source of switching noise because they are large in size and tend to switch simultaneously [2], [3]. Cases of switching noise degrading the performance of ADCs have also been reported [4]–[7].

Switching noise can couple to the analog signal path in several ways. For example, with a differential amplifier whose analog power supplies contain switching noise, an input-independent, additive noise component can appear in the differential output through device mismatch. Even with perfect matching, when the differential input is not zero, switching noise can appear at the output as a multiplicative noise component (i.e., the input is multiplied by scaled noise) [8] by coupling to the bias and modulating the small-signal parameters of the amplifier, such as the transconductance and output resistance or by varying the transistor's threshold voltage [9]. Also, with a differential sampling circuit that consists of transistor switches and sample capacitors, if switching noise is present in the substrate, this noise can couple to the circuit through parasitic capacitances that exist between the substrate and the top plates of the sample capacitors. If the differential paths are mismatched, an additive noise component appears at the output. Without the mismatch but with a nonzero differential input, a multiplicative noise component can appear at the output. The value of this multiplicative component depends on both the input and the noise [10] because the parasitic capacitance that links the substrate to the top plates of the sample capacitors contains junction capacitance that is input dependent.

A common method for mitigating the effects of switching noise has been to isolate the analog and digital circuits to minimize digital noise coupling into sensitive analog nodes. Isolation methods include large physical separation [11], the use of guard rings [12], [13], and the use of special process steps such as deep N-wells [2], [13]–[15]. While these methods can reduce the effects of switching noise coupling through the substrate, they increase die area and cost. Also, differential structures can reduce noise coupling effects by rejecting common-mode noise [3].

Finally, active noise minimization techniques have also been reported [9], [10], [16], [17]. In [10], negative feedback, substrate noise sensors, and dummy digital switches are used in a  $\Delta\Sigma$  loop to cancel substrate noise but are limited to use with

Manuscript received July 20, 2013; revised January 03, 2014, and March 05, 2014; accepted March 06, 2014. This paper was approved by Guest Editor Ichiro Fujimori. This work was supported in part by Analog Devices, Inphi, Linear Technology, Marvell Semiconductor, Northrop Grumman, the Agilent Foundation, the Broadcom Foundation, and NSF Grant 1016704. Chip fabrication was donated by TSMC.

N. C.-J. Chang was with the Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA, and is now with Marvell Semiconductor, Inc., Santa Clara, CA 95054 USA.

P. J. Hurst, B. C. Levy, and S. H. Lewis are with the Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA.

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

Digital Object Identifier 10.1109/JSSC.2014.2314446

TABLE I  
INVERTER OUTPUT  $w[m]$  TRANSITIONS AND THE CORRESPONDING  
NOISE VOLTAGE  $n[m]$

$w[m-1]$	$w[m]$	$n[m]$
0	0	0
0	1	$v_1$
1	0	$v_2$
1	1	0

$$|v_1| \neq |v_2| \neq 0$$

highly doped substrates. In [17], a spiral inductor feeds the detected noise signal on the digital power supply forward to an amplifier that produces an out-of-phase noise current to drive the substrate, but this technique is limited by the opamp bandwidth. To overcome these limitations, this paper presents a fully digital and background adaptive noise canceling technique that reduces the effects of switching noise coupling from digital buffers at the output of a pipelined ADC without noise sensors.

This paper is organized as follows. Section II briefly reviews the mechanism of switching noise generation. Section III shows a model of a pipelined ADC with switching noise. Section IV describes two algorithms that find and cancel the effects of switching noise due to digital output buffers. Section V presents the design of the prototype and its test modes. Then, measured results from the prototype chip are shown in Section VI, and finally, Section VII gives a conclusion.

## II. GENERATION OF SWITCHING NOISE

Standard CMOS logic has non-constant power supply and ground currents. With nonzero supply and ground inductances, these currents cause switching noise on the digital power supply and ground lines. This switching noise can inject into the substrate and often becomes the dominant source of switching noise [18]–[20]. Furthermore, if the power supply and ground are shared between analog and digital circuits, direct noise coupling occurs through the supplies.

Let  $n[m]$  denote the switching noise voltage sampled by an analog circuit at time index  $m$ . Table I shows the transitions of an inverter output  $w[m]$  and the corresponding  $n[m]$ , which is zero when the inverter's output is constant and nonzero otherwise. The magnitudes of the noise voltages  $v_1$  and  $v_2$  are not equal in general. Although every digital gate contributes switching noise, the largest gates contribute most of the noise in practice. In this work, the prototype includes large on-chip digital output buffers that are the dominant source of switching noise. For the remainder of this paper, these output buffers will be referred to as the *noise buffers*.

## III. PIPELINED ADC WITH SWITCHING NOISE

Fig. 1 shows a block diagram of a general pipelined ADC with  $K$  stages and with switching noise. As in a conventional pipelined ADC, each stage has an analog input and a low-resolution digital output  $D_1 \dots D_K$ . Also, each stage except the last produces an analog output  $r_1 \dots r_{K-1}$  that is the stage's amplified quantization error. However, unlike in a conventional

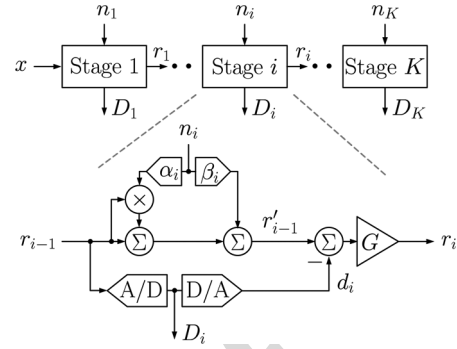


Fig. 1. Model of a pipelined ADC with switching noise. Scale factors  $\alpha_i$  and  $\beta_i$  are generally unknown. A multiplicative noise error is introduced by  $\alpha_i n_i$ , and an additive noise error is introduced by  $\beta_i n_i$ .

model, each stage also has a noise input  $n_1 \dots n_K$  to model the effect of undesired switching noise. An expanded view of stage  $i$  is also shown in Fig. 1.

As mentioned in Section I, switching noise  $n_i$  couples to the stage's bias voltages and supplies, introducing a multiplicative component that depends on the noise and the stage input. The model represents this effect as a multiplication in Fig. 1 of the analog residue  $r_{i-1}$  and the scaled noise  $\alpha_i n_i$ . The scale factor  $\alpha_i$  depends on processing and layout characteristics and is usually unknown. Switching noise  $n_i$  also couples to the analog input of the stage, introducing an additive component that is independent of the stage input. The model represents this effect as an addition in Fig. 1 where the scaled noise  $\beta_i n_i$  adds to the term  $r_{i-1}(1 + \alpha_i n_i)$ . Like  $\alpha_i$ , scale factor  $\beta_i$  is unknown. The result is the modified residue  $r'_{i-1}$ :

$$r'_{i-1} = r_{i-1} \underbrace{(1 + \alpha_i n_i)}_{\Delta g_i} + \underbrace{\beta_i n_i}_{\varepsilon_i}. \quad (1)$$

The term  $\alpha_i n_i$  in (1) is a *noise-induced gain error* denoted by  $\Delta g_i$  that multiplies  $r_{i-1}$ . A non-constant  $\Delta g_i$  introduces intermodulation tones in the ADC output by multiplying with the ADC input and stage residues. Because a residue contains a coarsely quantized version of the ADC input, it contains all harmonics of the input. As a result, the intermodulation tones in the ADC output appear at frequencies that are products of the harmonics and the noise. The general result of this effect has been shown in [8]. Moreover, a nonzero average of  $\Delta g_i$  scales the ADC input, causing an overall ADC gain error, while a nonzero average of  $\Delta g_i$  for  $i > 1$  introduces an interstage gain error between stages  $i-1$  and  $i$ , degrading the ADC's linearity. The term  $\beta_i n_i$  in (1) is an *additive noise error* denoted by  $\varepsilon_i$ . A nonzero  $\beta_i$  introduces an additive noise component in the ADC output that reduces the ADC's dynamic range without degrading linearity if the ADC has a sufficient correction range in every stage. Next, two switching noise cancellation algorithms are presented. One finds and cancels  $\Delta g_i$ , and the other finds and cancels  $\varepsilon_i$ .

## IV. ALGORITHMS FOR SWITCHING NOISE CANCELLATION

### A. Three-Stage Switching Noise Cancellation

Fig. 2 shows a block diagram of switching noise cancellation (SNC) at the output of the ADC. The ADC has a gain of 2 in

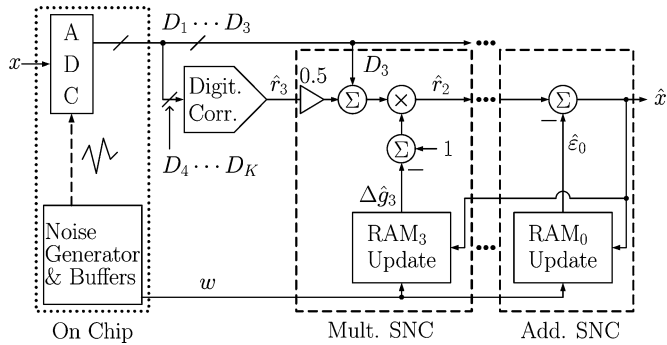


Fig. 2. 3-stage switching noise cancellation (SNC) performed off chip. Multiplicative SNC is used to cancel the noise-induced gain errors ( $\Delta g_i$ ) in the first three ADC stages ( $i = 1 \dots 3$ ). Additive SNC is used to cancel the sum of the input-referred additive noise errors ( $\varepsilon_0$ ).

each stage (i.e.,  $G = 2$  in Fig. 1), and the cancellation is done off chip using the digital outputs from each stage and the binary outputs  $w$  from the on-chip noise generator and buffers shown in the dotted box labeled as ‘On Chip’.

First, multiplicative SNC is performed in the first three stages because testing shows that SNC in these stages yields significant performance improvement. The number of stages is found empirically and was not predicted by simulation. The details for stage 3 are shown in the dashed box labeled as ‘Mult. SNC’. Let  $\hat{r}_3$  be the digitized residue output from stage 3, which is obtained after digital correction to remove redundancy from  $D_4 \dots D_K$ . Then,  $\hat{r}_3$  is scaled by 0.5 and added to  $D_3$  to obtain the digitized residue output from stage 2 before the noise error is corrected. In the analog part of the pipeline as shown in Fig. 1, for  $i = 3$ , an error was introduced by multiplying the analog residue  $r_2$  by  $1 + \Delta g_3$ , where  $\Delta g_3 = \alpha_3 n_3$ . [See (1).] This error can be corrected by dividing by  $1 + \Delta g_3$ . However, because division is difficult to implement in practice and  $\Delta g_3$  is unknown, this error is approximately corrected here by multiplying by  $1 - \Delta \hat{g}_3$ , where  $\Delta \hat{g}_3$  is an estimate of  $\Delta g_3$  if  $|\Delta g_3| \ll 1$ . The errors  $1 + \Delta g_1$  and  $1 + \Delta g_2$  in stages 1 and 2 are similarly corrected, but the block diagrams are not shown here for simplicity.

Next, additive SNC is performed. The details are shown in another dashed box labeled as ‘Add. SNC’. The additive noise terms from all the stages are canceled together through the term  $\hat{\varepsilon}_0$ , which is an estimate of the sum of the input-referred  $\varepsilon_i$  for all the stages, assuming the correction range is not exceeded in any stage. Under this condition, all interstage offsets and additive noise errors can be referred to the ADC input without introducing nonlinearity [21]. Then the new ADC output  $\hat{x}$  is used to update all the estimates of the noise-induced gain error  $\Delta \hat{g}_1 \dots \Delta \hat{g}_3$  and the estimate of the total additive noise error  $\varepsilon_0$ . The values of these estimates are stored in lookup tables implemented using RAMs as described in the next section.

The correction procedures described here do not correct for the higher-order terms that stem from interactions between  $\Delta g_i$  and  $\varepsilon_i$  in each stage (e.g.,  $\Delta g_i \varepsilon_j$  for  $i = j$  and  $i \neq j$ ). Testing reveals that these terms are below the resolution of the ADC in this work and can be ignored.

From an implementation standpoint, the delay from  $\hat{r}_3$  to  $\hat{x}$  needs to be less than one conversion cycle  $T$ , assuming the Digital Correction introduces no delay. Since three stages of SNC

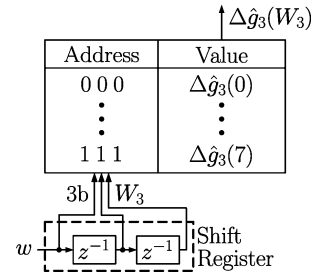


Fig. 3.  $RAM_3$  addressed by a vector formed by a shift register’s contents,  $W_3$ , which consists of the 3 most-recent outputs of the noise buffers  $w$  before the sampling instant of stage 3 in the ADC. Each period, an addressed value  $\Delta \hat{g}_3(W_3)$  is updated and used to compute a new  $\hat{x}$ .

are used in the prototype, three multipliers and four adders are present in the critical path. Ignoring the  $1/2$  scale operations between stages, each of these seven arithmetic functions needs to be completed within  $T/7$  or 3.6 ns for 40 MS/s conversion rate, assuming equal partitioning of the allowed delay.

### B. $RAM_3$ Lookup Table

Fig. 3 shows a diagram of a RAM used as a lookup table that stores the values of  $\Delta \hat{g}_3$ , and a shift register that addresses the RAM. The shift register input is  $w$ , which is the digital output of the noise buffers. The shift register contents form a vector of three one bit signals labeled as  $W_3$ , which addresses the RAM (i.e.,  $W_3 = \{w[m-2], w[m-1], w[m]\}$ ). The width of  $W_3$  is three bits because testing shows that the duration of the noise transient affecting a single stage is between two to three clock periods long. A longer transient requires a wider address, increasing the RAM size. When the noise buffers and the ADC are synchronous,  $W_3$  comes from the three most-recent outputs of the noise buffers before the sampling instant of stage 3, and each sample of  $W_3$  consists of 3 consecutive  $w$  values.

Each sample period, an addressed value  $\Delta \hat{g}_3(W_3)$  is read out of  $RAM_3$  to compute a new value of  $\hat{x}$ . Thus, the lookup table allows  $\Delta \hat{g}_3$  to be a general nonlinear function of  $W_3$ . One RAM lookup table corrects for the noise-induced gain error in one stage for one digital noise source. As a result, the total memory required for correction increases not only with the duration of the noise transient, but also with the number of significant and distinct digital noise sources. Since the outputs of a digital noise source (i.e.,  $w$ ) are known, noise sensors are not needed, and scaling in advanced technologies reduces the cost of integrated memory.

The use of a shift register here operating at the ADC conversion rate to detect the noise pattern is limited to noise that comes from a digital circuit operating at the same rate as the ADC. When the digital circuit operates at a faster rate than the ADC but is still synchronous with the ADC’s sample rate, the shift register needs to operate at the faster rate to capture the entire noise pattern. Then the shift register output can be down sampled to the ADC’s output rate so that the lookup table is updated at the same rate as the ADC output. If the noise comes from an asynchronous source, the noise pattern detected by the shift register does not provide enough information to determine the effect of the noise because asynchronous noise can occur at any time during the conversion period.

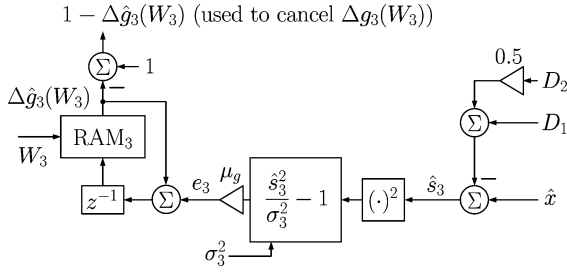


Fig. 4. Block diagram for updating  $\Delta\hat{g}_3$  (the estimate of the noise-induced gain error in stage 3), assuming  $\sigma_3^2$  (mean-square value of the input to stage 3) is known.

### C. To Update $\Delta\hat{g}_3$

Fig. 4 shows a block diagram that updates the estimate of the noise-induced gain error in stage 3,  $\Delta\hat{g}_3$ , using an adaptive maximum-likelihood method that follows a methodology developed by Bell and Sejnowski [22]. (See also [23], [24].) The algorithm assumes the ADC input  $x$  and the noise are independent, and the root-mean-square (RMS) noise is less than the RMS input.

The adaptation process begins with  $\hat{x}$ , which is the corrected ADC output. If  $\Delta\hat{g}_3$  is properly adjusted,  $\hat{x}$  contains no information about the noise-induced gain error in stage 3,  $\Delta g_3$ . However, if  $\Delta\hat{g}_3$  is not properly adjusted,  $\hat{x}$  contains information about that error. Furthermore, the most significant bits in  $\hat{x}$  produced before stage 3,  $D_1$  and  $D_2$ , do not give any information about  $\Delta g_3$ . To avoid interference that would delay the time when convergence would be reached,  $D_1$  and  $D_2$  are combined and subtracted from  $\hat{x}$ . After the subtraction, an estimate of  $s_3$ , which is the digitized input to stage 3, is produced as  $\hat{s}_3 = \hat{x} - (D_1 + D_2)/2$ . For stages 1 and 2,  $\hat{s}_1 = \hat{x}$  and  $\hat{s}_2 = \hat{x} - D_1$ , respectively. Next,  $\hat{s}_3$  is squared, and the squared value  $\hat{s}_3^2$  is divided by  $\sigma_3^2$ , which is the mean-square value of  $s_3$  or  $\overline{s_3^2}$ . For simplicity,  $\sigma_3^2$  is assumed to be a known constant here. Then 1 is subtracted from that ratio, and the result is scaled by a small factor  $\mu_g$ , producing

$$e_3 = \mu_g \left( \frac{\hat{s}_3^2}{\sigma_3^2} - 1 \right). \quad (2)$$

The signal  $e_3$  in (2) is used to update one location in  $\text{RAM}_3$  as determined by  $W_3$ , which is a vector of three consecutive outputs of the noise buffers as described in Section IV-B. Let  $e_3(W_3)$  denote the signal that is used to update  $\Delta\hat{g}_3(W_3)$  in  $\text{RAM}_3$ . For each  $W_3$  that addresses  $\text{RAM}_3$ , the average value of  $e_3(W_3)$  converges to 0 when the mean-square value of the corresponding samples in  $\hat{s}_3$  converges to  $\sigma_3^2$  (i.e.,  $\overline{e_3(W_3)} \rightarrow 0$  if  $\overline{\hat{s}_3^2(W_3)} \rightarrow \sigma_3^2$ ).

To update  $\Delta\hat{g}_3$ , the present value of  $\Delta\hat{g}_3(W_3)$  is summed with  $e_3(W_3)$ . The summer output is delayed and then written back to  $\text{RAM}_3$  in the same location addressed by  $W_3$  to complete the update. This whole structure sits in a negative feedback loop. The RAM, the summer and the delay form a code-dependent accumulator, which has infinite DC gain. The gain and the negative feedback work together to drive  $e_3(W_3)$  to 0 on average for all  $W_3$ . Then, the estimate  $\Delta\hat{g}_3$  overcomes the effect of  $\Delta g_3$ , and  $\overline{\Delta\hat{g}_3(W_3)} \approx \Delta g_3(W_3)$  if  $|\Delta g_3(W_3)| \ll 1$ .

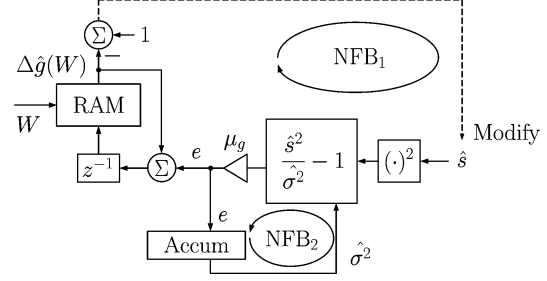


Fig. 5. Block diagram for updating  $\Delta\hat{g}$  and  $\hat{\sigma}^2$  (the estimate of the mean-square value of  $\hat{s}$ ) simultaneously. Subscript 3 is dropped here for simplicity.

In this section,  $\sigma_3^2$  is assumed known. However, in practice, it has to be found. So,  $\sigma_3^2$  is replaced by its estimate  $\hat{\sigma}_3^2$  next, and the conversion from  $\hat{x}$  to  $\hat{s}_3$  is ignored for simplicity.

### D. To Update $\hat{\sigma}^2$

Fig. 5 shows a simplified block diagram of the algorithm that updates  $\Delta\hat{g}$ . The subscript indicating stage 3 has been dropped for simplicity. The negative feedback loop from Fig. 4 is labeled here as NFB<sub>1</sub>. At the bottom of Fig. 5,  $\sigma^2 = \overline{s^2}$  is replaced by its estimate  $\hat{\sigma}^2$ . This estimate is found by a second negative feedback loop labeled as NFB<sub>2</sub>. This feedback loop uses each new value of  $e$  to update  $\hat{\sigma}^2$ . Hence, (2) can be rewritten as

$$e = \mu_g \left( \frac{\hat{s}^2}{\hat{\sigma}^2} - 1 \right). \quad (3)$$

Since the feedback is negative and the loop contains an accumulator (Accum) that operates on  $e$ , the average value of  $e$  is driven to 0. When that happens, the ratio  $\hat{s}^2/\hat{\sigma}^2$  is forced to converge to 1 on average. Then, as shown in the next section, when  $\hat{s}$  converges to its true value  $s$ ,  $\hat{\sigma}^2$  also converges to its true value  $\sigma^2$ . Because NFB<sub>1</sub> and NFB<sub>2</sub> interact with each other, convergence of these two loops is a concern. The next section gives a required condition for convergence.

### E. Convergence of $\Delta\hat{g}$ and $\hat{\sigma}^2$

This section gives an intuitive explanation of a condition required for convergence. First, the two loops NFB<sub>1</sub> and NFB<sub>2</sub> in Fig. 5 are coupled in the sense that  $\Delta\hat{g}$  depends on  $\hat{\sigma}^2$  and  $\hat{\sigma}^2$  depends on  $\Delta\hat{g}$ . Therefore, reducing the error in  $\Delta\hat{g}$  reduces the error in  $\hat{\sigma}^2$  and vice versa. A condition that allows one of the loops to sometimes work independently of the other is as follows. When the RAM address  $W$  contains either all 0's or all 1's, the output of the noise buffers has been constant, and no noise is present in  $\hat{s}$ . So,  $\Delta\hat{g}$  is not updated under this condition. (Instead,  $\Delta\hat{g}(0)$  and  $\Delta\hat{g}(7)$  are set to 0.) As a result,  $\hat{s} = s$  and NFB<sub>1</sub> is inactive in the noiseless cases, giving NFB<sub>2</sub> a chance to reduce the error in  $\hat{\sigma}^2$ , independent of NFB<sub>1</sub>.

To show the error in  $\hat{\sigma}^2$  is reduced when  $W$  contains either all 0's or all 1's, let  $\hat{\sigma}^2 = \sigma^2(1 + \hat{\rho})$ , where  $\hat{\rho}$  represents the error in  $\hat{\sigma}^2$  that is independent of  $s$  before convergence is reached. Also, let  $\hat{s} = s$ . Then (3) becomes

$$e = \mu_g \left( \frac{s^2}{\sigma^2(1 + \hat{\rho})} - 1 \right). \quad (4)$$



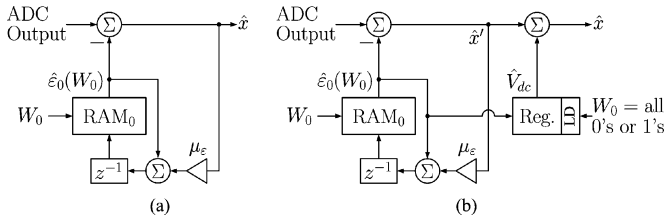


Fig. 6. Block diagram for updating  $\hat{\epsilon}_0$  (the estimate of the sum of the input-referred  $\epsilon_i$ ) (a) with and (b) without a DC null in  $\hat{x}$ . In (b), a register is used to store  $\hat{V}_{dc}$  (the estimate of the DC component at the ADC input). The register only loads (LD)  $\hat{V}_{dc}$  when  $W_0$  contains either all 0's or 1's.

Because  $\text{NFB}_2$  drives  $\bar{e} \rightarrow 0$  (Section IV-D),  $\overline{s^2}/(1 + \hat{\rho})/\sigma^2 \rightarrow 1$ , which can be approximated<sup>1</sup> by  $(\overline{s^2} - \overline{s^2\hat{\rho}}) \rightarrow \sigma^2$  if  $|\hat{\rho}| \ll 1$ . Because  $\overline{s^2} = \sigma^2$  by definition and  $\overline{s^2\hat{\rho}} = \overline{s^2} \cdot \overline{\hat{\rho}}$  under the assumption that  $s$  and  $\hat{\rho}$  are independent,  $\overline{\hat{\rho}} \rightarrow 0$  under the condition  $\hat{s} = s$ . Then, because  $\text{NFB}_1$  and  $\text{NFB}_2$  are coupled, reducing the error  $\hat{\rho}$  in  $\hat{\sigma}^2$  reduces the error in  $\Delta\hat{g}$ .

In summary, correct convergence requires three conditions: 1) The input and the noise are statistically independent. 2) The RMS noise is smaller than the RMS input. 3) The RAM address  $W$  sometimes contains either all 0's or all 1's (i.e.,  $\hat{s} = s$ ).

#### F. To Update $\epsilon_0$

Fig. 6(a) shows the adaption process for finding the estimate of the sum of the input-referred additive noise errors,  $\hat{\epsilon}_0$ . The subscript 0 indicates the error terms are summed and input referred. Testing reveals that cancellation in the first three ADC stages is sufficient and more stages yields insignificant additional improvement. Similar to  $W_3$  in Section IV-B,  $W_0$  is a vector that contains the four most-recent outputs of the noise buffers before the sampling instant of stage 3. Four bits are needed here because the total duration of the noise transients affecting the first three stages together is one clock period longer than for a single stage.

The estimate  $\hat{\epsilon}_0$  is found by a RAM-based LMS adaptation loop such as the one in [25], which can be viewed as an implementation and extension of the general concept of noise cancellation in [26]. The loop begins with the corrected ADC output  $\hat{x}$  and updates  $\hat{\epsilon}_0$  in  $\text{RAM}_0$  at a location addressed by  $W_0$  each sample period. When  $\hat{x}(W_0) \rightarrow 0$ ,  $\hat{\epsilon}_0$  overcomes the effect of  $\epsilon_0$ , and  $\hat{\epsilon}_0(W_0) \approx \epsilon_0(W_0)$  if the higher-order terms described at the end of Section IV-A are ignored.

If the ADC input  $x$  contains a nonzero DC component, all values in the RAM contain this component. Also, the ADC's input-referred offset  $V_{os}$  adds to the input at DC. Let  $V_{dc}$  denote the sum of the nonzero DC component in  $x$  and  $V_{os}$ . Because  $V_{dc}$  is in all RAM values and these values are subtracted from the ADC output, the algorithm introduces a DC null in the frequency spectrum of  $\hat{x}$ . To overcome this problem, a modified algorithm is used.

Fig. 6(b) shows the modification where  $\hat{x}'$  denotes the ADC output with a DC null. Also, the modification includes a new register and an extra summer. When  $W_0$  contains either all 0's

<sup>1</sup>The approximation here is done by using Taylor series expansion that  $1/(1 + \hat{\rho}) \approx 1 - \hat{\rho}$  if  $|\hat{\rho}| \ll 1$ . Because Taylor series expansion is valid on every sample of  $s$  and  $\hat{\rho}$ , it is also valid on the ensembles of  $s$  and  $\hat{\rho}$ .

TABLE II  
ESTIMATED DIE AREA AND POWER DISSIPATION OF THE SNC BLOCKS IN A  
0.18  $\mu\text{m}$  CMOS TECHNOLOGY

$f_S = 40 \text{ MS/s}$	Area ( $\text{mm}^2$ )	Power (mW)
Multiplicative SNC per stage	0.08	1.67
Additive SNC	0.09	0.39
Total for 3 stages	0.33	5.40

or all 1's, no digital noise occurs. Thus in these cases, the ADC output contains  $x(W_0) + V_{os}$  without noise, and the algorithm estimates  $V_{dc}$  by finding  $\hat{\epsilon}(W_0)$  after convergence. Let  $\hat{V}_{dc}$  denote the estimate of  $V_{dc}$ . A new value of  $\hat{V}_{dc}$  is loaded into the new register in Fig. 6(b) only during the noiseless cases. The value of  $\hat{V}_{dc}$  is read out of the register and added to every sample of  $\hat{x}'$  to produce a new  $\hat{x}$  that does not have a DC null. This technique removes the additive noise errors introduced by switching noise coupling, but it does not remove the ADC offset. To remove the offset, other calibration methods such as the one in [27] can be used.

#### G. Area and Power Dissipation Estimates

In practice, division in (3) can be avoided for simplicity by using  $e'$  rather than  $e$ , where

$$e' = \hat{\sigma}^2 e = \mu_g (\hat{s}^2 - \sigma^2). \quad (5)$$

This simplification makes the adaptation process in Section IV-D converge more slowly for the same step size  $\mu_g$  because  $e'$  in (5) is smaller than  $e$  in (3) for  $\hat{\sigma}^2 < 1$ . However, implementation with (5) avoids a numerical problem in (3) when  $\hat{\sigma}^2 = 0$ , and increasing  $\mu_g$  introduces more steady-state variation in  $e'$ . Starting with a large value of  $\mu_g$  and reducing  $\mu_g$  as the adaptation approaches convergence gives both fast convergence and small steady-state variation but was not done in the prototype. With this simplified implementation, Table II shows the estimated die area and power dissipation of the proposed 3-stage SNC blocks if implemented in a 0.18  $\mu\text{m}$  CMOS technology at a conversion rate  $f_S = 40 \text{ MHz}$ .

#### V. PROTOTYPE

Fig. 7(a) shows a block diagram of the floor plan of the prototype chip. The pipelined ADC's outputs are driven off chip through small drivers at the bottom of the figure. A bank of eight large and identical noise buffers whose gates are connected together is included at the top of the figure. A noise buffer is a digital output driver that consists of four tapered inverters. Furthermore, to ensure the noise buffers are the dominant source of switching noise, the ADC output is down-sampled on chip by a factor of six, reducing the required number of ADC drivers. Finally, by connecting the power supply of the noise buffers ( $V_{ddn}$ ) to the contacts of an N-well adjacent to the first pipeline stage, significant switching noise couples to the ADC via the substrate.

To demonstrate SNC, all the noise buffers are driven by the same input that is synchronous to the ADC through a multiplexer (MUX) that selects one of four noise sources: an external

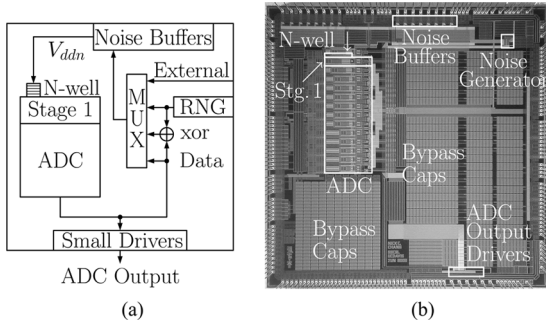


Fig. 7. Prototype chip's (a) floor plan and (b) die photograph. A noise buffer is a large digital output buffer. By connecting the power supply of the noise buffers ( $V_{ddn}$ ) to the contacts of an N-well adjacent to the first pipeline stage, significant switching noise couples to the ADC via the substrate.

source (External), a random number generator output (RNG), an ADC output without scrambling (Data), or the same ADC output with scrambling (Data xor RNG) [28]. The ADC output used for noise coupling is a raw digital output from the first stage in a 1.5 bit per stage pipeline. This binary output is 1 when the ADC input is greater than the comparator threshold  $V_{ref}/4$  and 0 otherwise, where  $\pm V_{ref}$  denotes the input full-scale voltage of the ADC. Scrambling is done by 'XORing' this ADC output bit with a random binary signal. Finally, the external source generates a square wave to provide a periodic noise pattern. Fig. 7(b) shows the corresponding die photograph of the prototype chip, which was fabricated in a  $0.18 \mu\text{m}$  CMOS process on a lightly-doped substrate. The area of the chip is  $25 \text{ mm}^2$ . The pipelined ADC consists of 12 1.5 bit stages followed by a 2 bit flash converter. The active area of the ADC is  $3 \text{ mm}^2$ . The bank of noise buffers is on the top side of the chip. The locations of the first pipeline stage and the noise-injected N-well are also shown.

## VI. MEASURED RESULTS

All measured results come from the simplified implementation described in Section IV-G. The conversion rate of the prototype is limited by the settling time of the interstage amplifiers in the pipelined ADC and is set to 40 MS/s in the tests below except where stated otherwise.

### A. Periodic Noise

Fig. 8 shows three integral-nonlinearity plots: without switching noise (top), with periodic switching noise injected into the N-well before SNC (middle), and with this noise after SNC (bottom). In this paper, all the code density tests [29] have a resolution of 12 bits and use an input amplitude that covers about 90% of the full-scale range. The other 10% of the range is saved to accommodate the switching noise without exceeding the full-scale range. A similar approach has been used previously when dither has been added to the input [30]. Without switching noise, the noise buffers are inactive, and the maximum magnitude of the INL in the top plot of Fig. 8 is about 0.7 LSB. With periodic noise, the maximum INL in the middle plot increases to 2.3 LSB. The main observation here is that the INL pattern resembles the pattern caused by interstage gain errors. This result stems from the fact that noise-induced

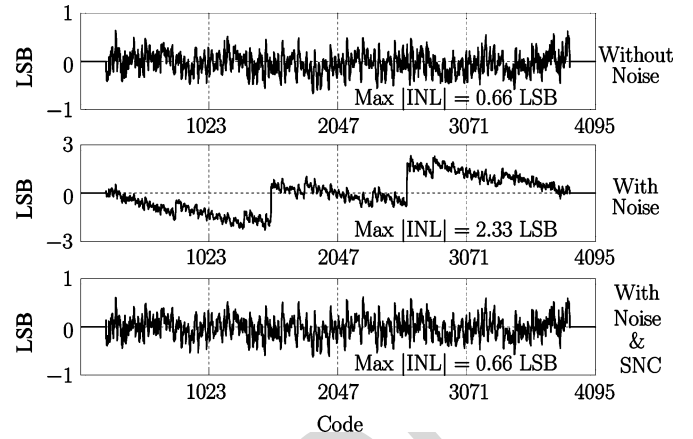


Fig. 8. INL plots related to periodic noise.

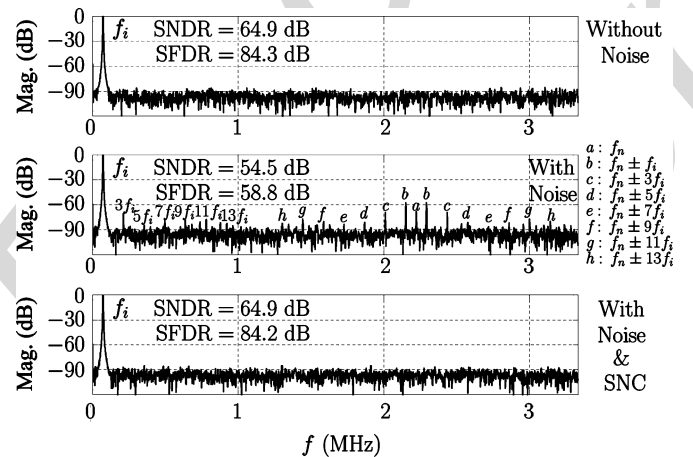


Fig. 9. FFT plots related to periodic noise.

gain error changes the effective gain applied to the stage input. Also, the middle INL plot has a downward trend in the regions from code 0 to 1500, code 1500 to 2500, and code 2500 to 4000 with vertical jumps upward at codes 1508 and 2591. This downward trend shows the effective interstage gain between the first and second stages is greater than the ideal value of 2 due to the switching noise. This effective gain error corresponds to the average value of  $\Delta g_2$  because  $\Delta g_2$  scales the analog residue output from stage 1 (and  $\Delta g_1$  scales the ADC input), as described in Section III. After SNC, the maximum INL in the bottom plot of Fig. 8 is about the same as in the case without noise.

Fig. 9 shows three FFT plots: without switching noise, with periodic switching noise before SNC, and with this noise after SNC. In all three cases, the conversion rate  $f_S = 40 \text{ MS/s}$ , and the input frequency  $f_i = 71 \text{ kHz}$ . Also, because the output is down-sampled on chip by a factor of 6, the x-axes on the plots extend to half the down-sampled conversion rate,  $1/2 \times f_S/6 \approx 3.3 \text{ MHz}$ . Since the ADC output is down-sampled in this project, the noise cancellation (i.e., multiplication and subtraction to compute  $\hat{x}$  in Fig. 2) is performed at the down-sampled rate, and the RAM updates are also performed at that rate. Without switching noise as shown in the top plot of Fig. 9, the SNDR and SFDR are 64.9 dB and 84.3 dB, respectively. With

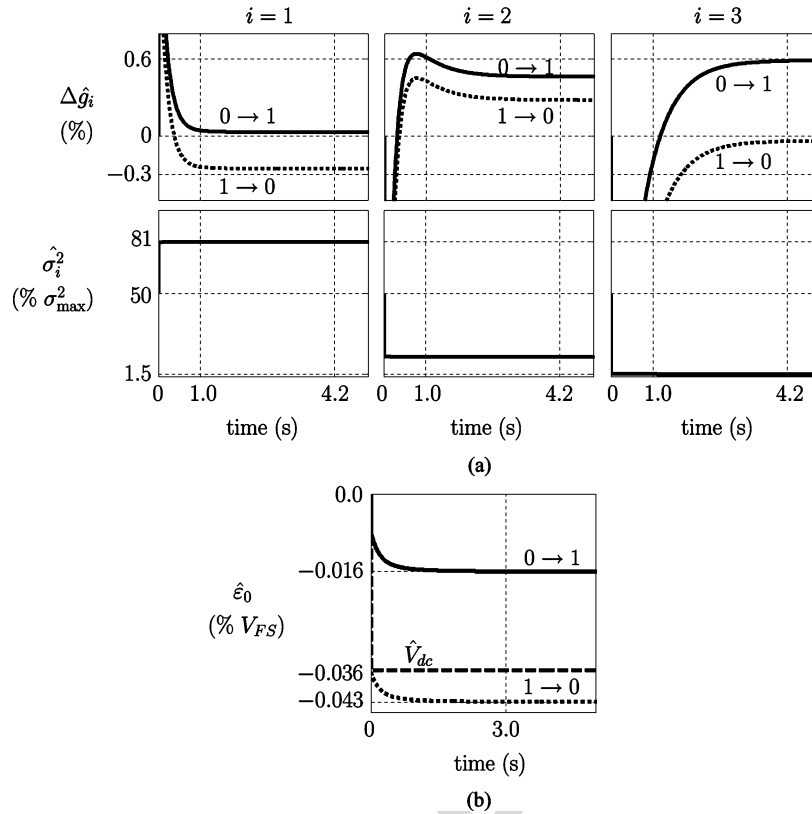


Fig. 10. Measured plots for (a)  $\Delta\hat{g}_i$  and  $\hat{\sigma}_i^2$ , and (b)  $\hat{\varepsilon}_0$  versus time for the periodic-noise case. The plots labeled as  $0 \rightarrow 1$  and  $1 \rightarrow 0$  show values of the noise estimates that correspond to switching the noise buffers from 0 to 1 and 1 to 0, respectively, right before the sampling instant of the first ADC stage.

all eight noise buffers turned on, switching noise appears and the SNDR and SFDR degrade to 54.5 dB and 58.8 dB, respectively. The switching noise here is generated by switching the noise buffers with a square wave at a frequency  $f_n$ , which is 1/3 of the down-sampled conversion rate,  $(1/3) \times (f_s/6) \approx 2.2$  MHz. Although higher-order noise tones may be hidden in the FFT plot by the choice of  $f_n$ , the chosen frequency allows for large noise coupling to the ADC after down-sampling. Because  $f_n$  is 1/3 of the down-sampled conversion rate, 2/3 of the down-sampled ADC outputs are affected by  $0 \rightarrow 1$  and  $1 \rightarrow 0$  transitions of the square-wave noise. The remaining 1/3 of the ADC outputs correspond to the noiseless samples required by the convergence condition specified in Section IV-E. With this noise, the main difference between the two top plots appears at frequencies labeled as  $a, b, \dots, h$ . The tone at  $a$  stems from additive noise and appears at  $f_n = 2.2$  MHz. The tones at  $b, c, \dots, h$  stem from intermodulation between the harmonics of the input at  $f_i$  and the switching noise at  $f_n$  (e.g.,  $f_n \pm f_i$  and  $f_n \pm 3f_i$ ). Also with noise, input harmonic tones (e.g.,  $3f_i$ ) appear because switching noise causes interstage gain errors as explained with the INL plots in Fig. 8. After SNC, the intermodulation tones and other undesired noise components are suppressed as shown in the bottom plot of Fig. 9. The SNDR and SFDR in the bottom case become 64.9 dB and 84.2 dB, respectively. SNC suppresses periodic noise in the prototype generated by noise buffers operating synchronously with the ADC for all frequencies up to half the sampling rate under the assumptions stated in Section IV-E.

The top of Fig. 10(a) shows measured plots for the estimates of the noise-induced gain errors in the first three stages,  $\Delta\hat{g}_i$ ,

versus time, where  $i$  indicates stage 1, 2, and 3. The values of  $\Delta\hat{g}_i$  are plotted in % and range from  $-0.3\%$  to  $+0.6\%$  after convergence. For each stage, the solid curve corresponds to switching the noise buffers from  $0 \rightarrow 1$  right before the sampling instant of the first ADC stage, and the dotted curve corresponds to  $1 \rightarrow 0$  transitions at the same instant.

The bottom of Fig. 10(a) shows the corresponding plots for the estimates of the mean-square values of the digitized stage inputs,  $\hat{\sigma}_i^2$  for  $i = 1, 2, 3$ , versus time. The values of these estimates are plotted in % of  $\sigma_{\max}^2$ , which is the mean-square value of a sinusoidal input that covers the full-scale input range. The values of these estimates range from 1.5% to 81% after convergence.

In general, the convergence time of  $\Delta\hat{g}_i$  depends mainly on the occurrence of noise samples at the ADC output; convergence is faster when noise samples occur more frequently. Also,  $\hat{\sigma}_i^2$  converges faster than  $\Delta\hat{g}_i$  because  $\hat{\sigma}_i^2$  is always updated with every sample of  $e$  in (3), and  $\Delta\hat{g}_i$  is only updated when a noise transition occurs.

Fig. 10(b) shows three measured plots for the estimate of the input-referred additive noise error  $\hat{\varepsilon}_0$  versus time. The first two plots are for  $\hat{\varepsilon}_0$ , where the solid curve is for  $0 \rightarrow 1$  transitions right before the sampling instant of the first ADC stage, and the dotted curve is for  $1 \rightarrow 0$  transitions at the same instant. Both plots include the estimated DC component  $\hat{V}_{dc}$  at the ADC input (about  $-0.036\%$  of full scale). The third plot is dashed and shows  $\hat{V}_{dc}$ . The values of  $\hat{\varepsilon}_0$  are plotted in % of full scale and range from  $-0.016\%$  to about  $-0.043\%$  after convergence.

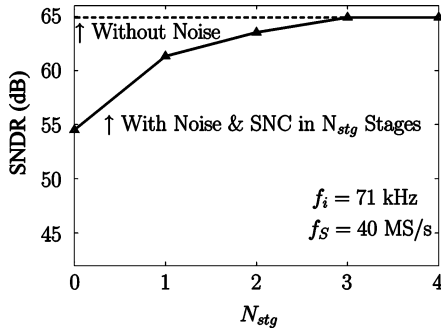


Fig. 11. SNDR versus the number of stages  $N_{stg}$  in which SNC is applied, for the periodic-noise case.

This estimate converges within 3 seconds for  $f_S = 40$  MS/s and for a step size  $\mu_\varepsilon = 2^{-14}$ .

In the periodic-noise case, the effective noise-induced gain error  $\Delta g_2$  can be calculated from Fig. 10(a). Both  $0 \rightarrow 1$  and  $1 \rightarrow 0$  transitions affect the ADC output once every three samples because  $f_n$  is  $1/3$  of the down-sampled conversion rate. From Fig. 10(a), the estimated values of  $\Delta g_2$  for  $0 \rightarrow 1$  and  $1 \rightarrow 0$  transitions are about 0.45% and 0.30%, respectively. Thus their average value is about  $(0.45\% + 0.30\%) / 3 = 0.25\%$ , which is about the same as the calculated  $\Delta g_2$  inferred from the measured INL plot ( $\sim 0.24\%$ ).

Fig. 11 plots the SNDR versus the number of stages  $N_{stg}$  with SNC. When  $N_{stg} = 1$ , only the first stage is corrected. When  $N_{stg} = 3$ , the first three stages are all corrected, and the SNDR is the same as without noise; therefore, no further correction is needed. An overall improvement of 10 dB is seen. About 7 dB of this improvement comes from correcting just the first stage.

### B. Random Noise

Fig. 12 shows three FFT plots: without deliberately injected switching noise, with switching noise that comes from the random number generator before SNC, and with this random noise after SNC. Again,  $f_i = 71$  kHz. Without switching noise, the noise buffers are turned off and the SNDR and SFDR are 64.9 dB and 84.3 dB, respectively. With all eight noise buffers driven by the same random noise, the noise floor is raised, degrading the SNDR to 51.9 dB (3 dB worse than with periodic noise). Also, the SFDR falls to 75.3 dB, which is about 17 dB higher than with periodic noise because the random noise is white, unlike the periodic noise. Since the noise is white, every switching pattern,  $W = 000, \dots, 111$ , is equally likely. So, each pattern occurs one in eight samples on average, and a noiseless pattern ( $W = 000$  or  $W = 111$ ) occurs one in four samples on average. Finally, SNC reduces the noise floor and restores the SNDR and SFDR to nearly their levels without switching noise.

Fig. 13 shows plots of SNDR and SFDR versus  $f_S$  without and with periodic and random noise for conversion rates from 5 MS/s to 40 MS/s. Each case shows three curves. The solid curve represents the case without noise. The dashed curve represents the case with noise but without SNC. The dotted curve just below the solid curve represents the case with noise and with SNC. The ADC input is sinusoidal at  $f_i = 71$  kHz in all cases. Turning on the noise without SNC introduces a smaller

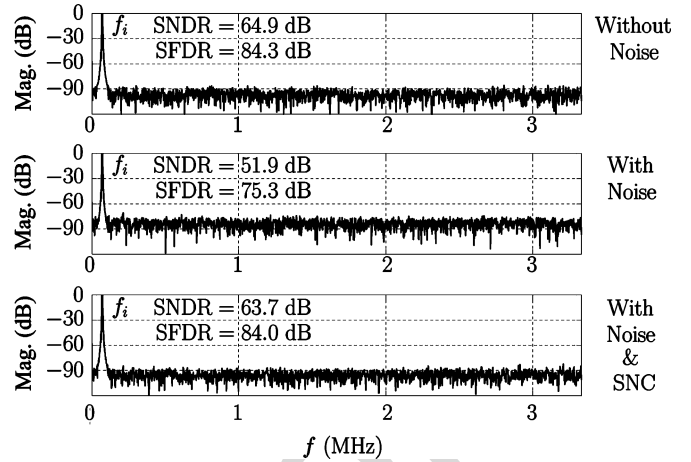


Fig. 12. FFT plots related to random noise.

penalty at 5 MS/s than at 40 MS/s. For example, in Fig. 13(a), the lower, dashed plot with noise but without SNC shows the SNDR at 5 MS/s is about 13 dB higher than at 40 MS/s. Also, in Fig. 13(b), the SFDR at 5 MS/s is about 15 dB higher than at 40 MS/s. The smaller penalties stem from the fact that the transients introduced by the noise have enough time to decay to insignificant levels between samples at 5 MS/s. Furthermore, in Figs. 13(a) and (b), the SNDR and SFDR at 20 MS/s in the dashed plot are about 3 dB worse than at 40 MS/s. This higher penalty at a lower frequency shows that the noise transients do not always add constructively with increasing frequency. Finally, for the periodic-noise case, turning on SNC just about restores the performance to the original levels. For the random-noise case, turning on SNC restores the performance within 2 dB of the original levels.

The periodic-noise case converges faster than the random-noise case (e.g.,  $\sim 120$  k samples versus  $\sim 1$  M samples) because periodic noise always drives the error function (3) in the same direction until convergence is achieved. In the random-noise case, convergence requires more samples to average out the random fluctuation in the error function, unlike the periodic-noise case.

### C. ADC Output Noise With Scrambling

For this case, the noise is caused by an output bit from the first ADC stage with scrambling that drives all 8 noise buffers. Scrambling is required because the noise is otherwise input dependent, and the SNC algorithm is not effective in that case. The noise without and with scrambling will be referred to as *unscrambled* and *scrambled noise*, respectively. The scrambled noise does not generate harmonic tones, but the noise floor in the FFT plot (not shown for brevity) rises. This effect stems from a whitening of the noise floor provided by scrambling, similar to the random-noise case shown in Section VI-B. Also, because the ADC output bit that drives the noise buffers changes so slowly that insignificant noise is generated with  $f_i = 71$  kHz,  $f_i$  is increased to 9.112 MHz here. This input frequency is about 100 times higher than in the previous cases and is chosen so that the performance degradation with noise and without SNC is similar to that in the periodic-noise case.

TABLE III  
PERFORMANCE SUMMARY

Resolution	12 bits									
Conversion Rate $f_S$	40 MS/s									
Technology	0.18- $\mu$ m CMOS, Lightly-doped									
Supply Voltages	1.8 V (ADC and Noise Generator), 3.3 V (Noise Buffers and ADC Output Buffers)									
Full-Scale $V_{FS}$	1.6 V <sub>p-p</sub> Differential									
Active Area	3 mm <sup>2</sup> (ADC), 1 mm <sup>2</sup> (Noise Generator & Buffers)									
Power	99 mW (Analog), 17 mW (Digital)									
Input Frequency $f_i$	71 kHz					9.112 MHz				
Noise Type	None	Periodic		Random		None	Unscrambled <sup>+</sup>		Scrambled <sup>+</sup>	
SNC	No	No	Yes	No	Yes	No	No	Yes	No	Yes
INL [LSB]	0.66	2.33	0.66	1.74	0.69	0.66	1.72	n/a	1.64	0.78
DNL [LSB]	0.26	0.67	0.26	0.50	0.31	0.24	0.45	n/a	0.45	0.25
SNDR [dB]	64.9	54.5	64.9	51.9	63.7	63.4	55.8	n/a	51.8	62.1
SFDR [dB]	84.3	58.8	84.2	75.3	84.0	83.3	59.7	n/a	70.6	83.1
ENOB [bits]	10.5	8.8	10.5	8.3	10.3	10.2	9.0	n/a	8.3	10.0

<sup>+</sup> ADC output without or with scrambling.

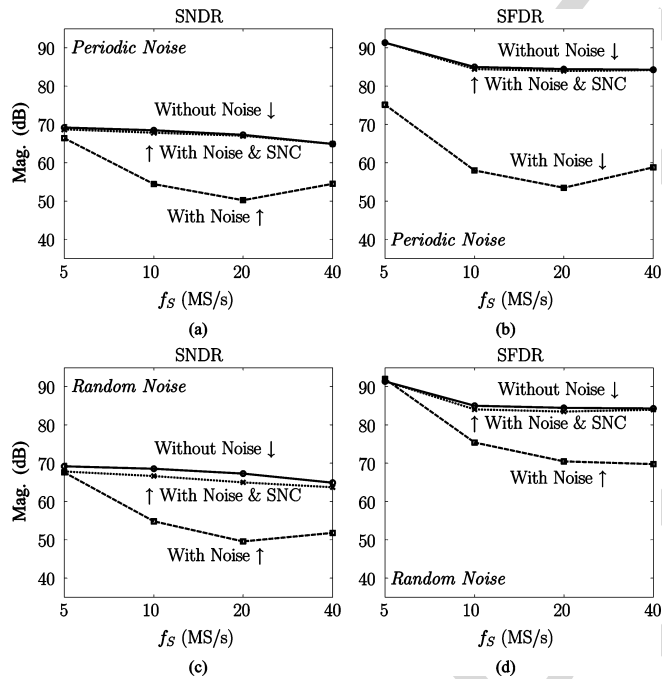


Fig. 13. SNDR and SFDR versus conversion rate  $f_S$  for two noise cases: (a)-(b) periodic noise and (c)-(d) random noise.  $f_i = 71$  kHz.

Fig. 14 shows four plots of SNDR versus  $f_i$  in log scale for  $f_S = 40$  MS/s. The x-axis extends to the Nyquist frequency, 20 MHz. The solid curve is without noise. The SNDR in that case drops after about 10 MHz because of jitter on the

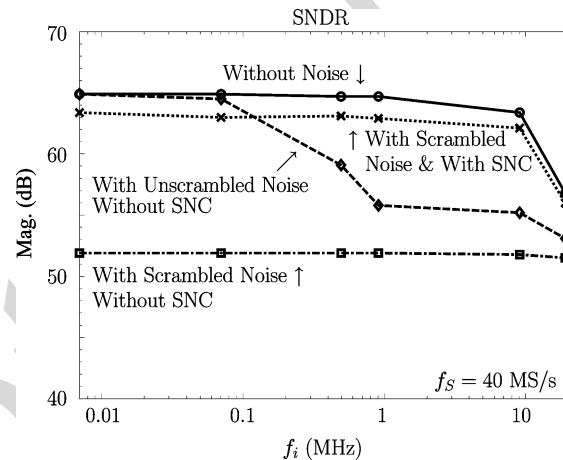


Fig. 14. SNDR versus input frequency  $f_i$  for the ADC output noise without and with scrambling.  $f_S = 40$  MHz.

sample clock. The other three cases are with noise. The dashed curve is with unscrambled noise and without SNC. The SNDR falls monotonically with increasing  $f_i$  in this case. This trend stems from the fact that increasing  $f_i$  causes the ADC outputs to change more frequently and introduce more noise, degrading the SNDR. Since SNC is not helpful without scrambling, the case with unscrambled noise and with SNC is omitted in Fig. 14. The last two cases are with scrambled noise. The case without SNC is at the bottom of Fig. 14 in the dash-dot curve, and the case with SNC is second from the top in the dotted curve. With scrambled noise and without SNC, the level of noise does not

depend on the input frequency  $f_i$ . In this case, the SNDR is about the same as with random noise, which is about 11 dB down from the case without noise. With scrambled noise and with SNC, the SNDR is within 2 dB of the case without noise.

## VII. CONCLUSION

Table III summarizes performance. This work shows that the additive and multiplicative noise errors caused by switching noise in a pipelined ADC can be found and canceled at the output using a digital background adaptive technique. The proposed technique does not require noise sensors, and this technique is mainly useful when the noise is dominated by the activity of a small number of digital signals. In practice, this technique may be combined with the traditional strategies such as guard banding and physical separation to minimize the cost of noise reduction in area and power dissipation.

## REFERENCES

- [1] K. Bult, "Broadband communication circuits in pure digital deep sub-micron CMOS," in *IEEE Int. Solid-State Circuits (ISSCC) Dig. Tech. Papers*, 1999, pp. 76–77.
- [2] M. Badaroglu *et al.*, "Modeling and experimental verification of substrate noise generation in a 220-Kgates WLAN system on-chip with multiple supplies," *IEEE J. Solid-State Circuits*, vol. 38, no. 7, pp. 1250–1260, Jul. 2003.
- [3] K.-S. Ha *et al.*, "A 0.13- $\mu\text{m}$  CMOS 6 Gb/s/pin memory transceiver using pseudo-differential signaling for removing common-mode noise due to SSN," *IEEE J. Solid-State Circuits*, vol. 44, no. 11, pp. 3146–3162, Nov. 2009.
- [4] B. Brandt and B. Wooley, "A 50-MHz multibit Sigma-Delta modulator for 12-b 2-MHz A/D conversion," *IEEE J. Solid-State Circuits*, vol. 26, no. 12, pp. 1746–1756, Dec. 1991.
- [5] M. Ito *et al.*, "A 10 bit 20 MS/s 3 V supply CMOS A/D converter," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1531–1536, Dec. 1994.
- [6] T. Blalack and B. Wooley, "The effects of switching noise on an oversampling A/D converter," in *IEEE Int. Solid-State Circuits (ISSCC) Dig. Tech. Papers*, 1995, pp. 200–201, 367.
- [7] M. Mayes and S. W. Chin, "A 200 mW, 1 MS/s, 16-b pipelined A/D converter with on-chip 32-b microcontroller," *IEEE J. Solid-State Circuits*, vol. 31, no. 12, pp. 1862–1872, Dec. 1996.
- [8] M. Xu *et al.*, "Measuring and modeling the effects of substrate noise on the LNA for a CMOS GPS receiver," *IEEE J. Solid-State Circuits*, vol. 36, no. 3, pp. 473–485, Mar. 2001.
- [9] X. Aragones *et al.*, *Analysis and Solutions for Switching Noise Coupling in Mixed-Signal ICs*, 1st ed. Delft, The Netherlands: Kluwer Academic, 1999.
- [10] M. Peng and H.-S. Lee, "Study of substrate noise and techniques for minimization," *IEEE J. Solid-State Circuits*, vol. 39, no. 11, pp. 2080–2086, Nov. 2004.
- [11] P. van Zeijl *et al.*, "A Bluetooth radio in 0.18- $\mu\text{m}$  CMOS," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1679–1687, Dec. 2002.
- [12] C. Soens *et al.*, "Modeling of substrate noise generation, isolation, and impact for an LC-VCO and a digital modem on a lightly-doped substrate," *IEEE J. Solid-State Circuits*, vol. 41, no. 9, pp. 2040–2051, Sep. 2006.
- [13] W.-K. Yeh *et al.*, "Substrate noise-coupling characterization and efficient suppression in CMOS technology," *IEEE Trans. Electron Devices*, vol. 51, no. 5, pp. 817–819, May 2004.
- [14] A. Ali *et al.*, "A 14 bit 125 MS/s IF/RF sampling pipelined ADC with 100 dB SFDR and 50 fs jitter," *IEEE J. Solid-State Circuits*, vol. 41, no. 8, pp. 1846–1855, Aug. 2006.
- [15] R. Rossi *et al.*, "Model and verification of triple-well shielding on substrate noise in mixed-signal CMOS ICs," in *Proc. 29th ESSCIRC*, 2003, pp. 643–646.
- [16] K. Makie-Fukuda *et al.*, "Substrate noise reduction using active guard band filters in mixed-signal integrated circuits," in *Symp. VLSI Circuits Dig. Tech. Papers*, 1995, pp. 33–34.
- [17] T. Nakura *et al.*, "Feedforward active substrate noise canceling technique using power supply di/dt detector," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2005, pp. 284–287.
- [18] M. Nagata *et al.*, "Effects of power-supply parasitic components on substrate noise generation in large-scale digital circuits," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2001, pp. 159–162.
- [19] M. van Heijningen *et al.*, "Analysis and experimental verification of digital substrate noise generation for epi-type substrates," *IEEE J. Solid-State Circuits*, vol. 35, no. 7, pp. 1002–1008, Jul. 2000.
- [20] G. Van der Plas *et al.*, "High-level simulation of substrate noise in high-ohmic substrates with interconnect and supply effects," in *Proc. 41st Design Automation Conf.*, 2004, pp. 854–859.
- [21] S. Lewis and P. Gray, "A pipelined 5-MS/s 9 bit analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. SSC-22, no. 6, pp. 954–961, Dec. 1987.
- [22] A. J. Bell and T. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, pp. 1129–1159, 1995.
- [23] D. J. MacKay, "Maximum Likelihood and Covariant Algorithms for Independent Component Analysis," Univ. Cambridge, Cambridge, U.K., Tech. Rep., 1996.
- [24] J.-F. Cardoso, "Infomax and maximum likelihood for blind source separation," *IEEE Signal Process. Lett.*, vol. 4, no. 4, pp. 112–114, Apr. 1997.
- [25] K. Fisher *et al.*, "An adaptive RAM-DFE for storage channels," *IEEE Trans. Commun.*, vol. 39, no. 11, pp. 1559–1568, Nov. 1991.
- [26] B. Widrow *et al.*, "Adaptive noise canceling: Principles and applications," *Proc. IEEE*, vol. 63, no. 12, pp. 1692–1716, Dec. 1975.
- [27] S. Jamal *et al.*, "A 10-b 120-MS/s time-interleaved analog-to-digital converter with digital background calibration," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1618–1627, Dec. 2002.
- [28] R. Jewett *et al.*, "A 12b 128 MS/s ADC with 0.05 LSB DNL," in *IEEE Int. Solid-State Circuits (ISSCC) Dig. Tech. Papers*, 1997, pp. 138–139, 443.
- [29] J. Doernberg *et al.*, "Full-speed testing of A/D converters," *IEEE J. Solid-State Circuits*, vol. SSC-19, no. 6, pp. 820–827, Dec. 1984.
- [30] J. Ming and S. Lewis, "An 8 bit 80-MS/s pipelined analog-to-digital converter with background calibration," *IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1489–1497, Oct. 2001.



**Nick C.-J. Chang** received the B.S. and M.S. degrees from Columbia University, New York, NY, USA, and the Ph.D. degree from the University of California, Davis, CA, USA, all in electrical engineering.

In 2011, he joined Marvell Semiconductor, where he built an ADC for an Ethernet transceiver. His research interests include data conversion technique, digital signal processing, and analog circuit design.



**Paul J. Hurst** (S'76–M'83–SM'94–F'01) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of California, Berkeley, CA, USA, in 1977, 1979, and 1983, respectively.

From 1983 to 1984, he was with the University of California, Berkeley, as a lecturer, teaching integrated-circuit design courses and working on an MOS delta-sigma modulator. In 1984, he joined Silicon Systems Inc., Nevada City, CA, USA, where he was involved in the design of CMOS integrated circuits for voice-band modems. Since 1986, he has

been on the faculty of the Department of Electrical and Computer Engineering at the University of California at Davis, where he is now a Professor. His research interests are in the areas of data converters and analog and mixed-signal integrated-circuit design for digital communications. He is a co-author of a textbook on analog integrated-circuit design. He is also active as a consultant to industry.

Prof. Hurst has served on the program committees for the Symposium on VLSI Circuits and the IEEE International Solid-State Circuits Conference. He has served as an associate editor for the IEEE JOURNAL OF SOLID-STATE CIRCUITS and as a member of the administrative committee of the IEEE Solid-State Circuits Society.



**Bernard C. Levy** received the diploma of Ingénieur Civil des Mines from the Ecole Nationale Supérieure des Mines in Paris, France, in 1974, and the Ph.D. degree in electrical engineering from Stanford University, Stanford, CA, USA, in 1979.

From July 1979 to June 1987, he was an Assistant and then Associate Professor in the Department of Electrical Engineering and Computer Science at Massachusetts Institute of Technology (M.I.T.), Cambridge, MA, USA. Since July 1987, he has been with the University of California, Davis, CA, USA,

where he is Professor of Electrical Engineering and a member of the Graduate Group in Applied Mathematics. He served as Chair of the Department of Electrical and Computer Engineering at UC Davis from 1996 to 2000. He was a Visiting Scientist at the Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA) in Rennes, France from January to July 1993, and at the Institut National de Recherche en Informatique et Automatique (INRIA), in Rocquencourt, France, from September to December 2001. He is the author of the book *Principles of Signal Detection and Parameter Estimation* (Springer, 2008). His research interests include statistical signal processing, estimation, detection, and circuits applications of signal processing.

Dr. Levy served as an Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS I, the IEEE TRANSACTIONS ON CIRCUITS AND

SYSTEMS II, and the EURASIP *Journal on Advances in Signal Processing*. He is currently an Associate Editor of *Signal Processing*. He is a member of SIAM.



**Stephen H. Lewis** (S'85–M'88–SM'97–F'01) received the B.S. degree from Rutgers University, New Brunswick, NJ, USA, in 1979, the M.S. degree from Stanford University, Stanford, CA, USA, in 1980, and the Ph.D. degree from the University of California, Berkeley, CA, USA, in 1987, all in electrical engineering.

From 1980 to 1982, he was with Bell Laboratories, Whippany, NJ, USA, where he was involved in circuit design for magnetic recording. In 1988, he re-joined Bell Laboratories, Reading, PA, USA, where he concentrated on the design of analog-to-digital converters. In 1991, he joined the Department of Electrical and Computer Engineering, University of California, Davis, CA, USA, where he is now a Professor. He is a co-author of a college textbook on analog integrated circuits, and his research interests include data conversion, signal processing, and analog circuit design.

IEEE Pre-proof  
Print Version