A 12-Bit 20-Msample/s Pipelined Analog-to-Digital Converter With Nested Digital Background Calibration

Xiaoyue Wang, *Member, IEEE*, Paul J. Hurst, *Fellow, IEEE*, and Stephen H. Lewis, *Fellow, IEEE*

***Abstract*—A 12-bit 20-Msample/s pipelined analog-to-digital converter (ADC) is calibrated in the background using an algorithmic ADC, which is itself calibrated in the foreground. The overall calibration architecture is nested. The calibration overcomes the circuit nonidealities caused by capacitor mismatch and finite operational amplifier (opamp) gain both in the pipelined ADC and the algorithmic ADC. With a 58-kHz sinusoidal input, test results show that the pipelined ADC achieves a peak signal-to-noise-and-distortion ratio (SNDR) of 70.8 dB, a peak spurious-free dynamic range (SFDR) of 93.3 dB, a total harmonic distortion (THD) of −92.9 dB, and a peak integral nonlinearity (INL) of 0.47 least significant bit (LSB). The total power dissipation is 254 mW from 3.3 V. The active area is 7.5 mm^2 in 0.35-μm CMOS.**

***Index Terms*—Analog-to-digital conversion, digital background calibration, nested calibration, CMOS analog integrated circuits.**

I. Introduction

BACKGROUND calibration improves analog-to-digital converter (ADC) linearity without interrupting the input conversion. When background calibration is carried out in the digital domain, it can take advantage of scaling that stems from Moore's law. Previously used background-calibration techniques have: 1) linearized digital-to-analog subconverters in multistage ADCs [1]–[3]; 2) limited the input signal bandwidth below half the sampling rate [4], [5]; 3) reduced the input dynamic range and/or the correction range (the amount of decision-level movement that can be tolerated without error) [2], [3], [6]–[10]; 4) used analog techniques that do not readily scale to new process technologies [8], [11]; or 5) required more than one clock frequency [12], [13]. In principle, these limitations can be overcome with a new architecture that compares the output of the ADC under calibration to that of a reference ADC. However, the linearity of the reference ADC limits the accuracy with this approach. To overcome this limitation, the reference ADC can itself be calibrated to achieve the required linearity [14].

The result is a fast and accurate ADC that does not require any high-gain opamps. This is important because scaling in CMOS process technologies is reducing power-supply voltages and the gains of single-stage opamps, which are faster than multistage opamps for a given power dissipation. The cost of the calibration is increased complexity in the digital domain, but scaling predicted by Moore's law is dramatically reducing the area and power dissipation of the digital processing, making the ADC architecture described in this paper compatible with the characteristics of modern CMOS processes.

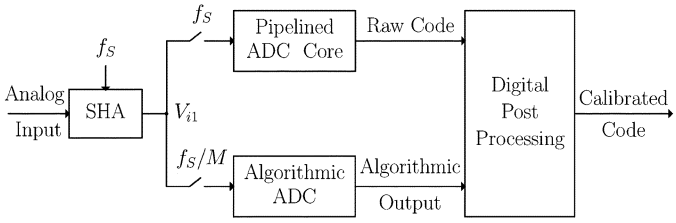


Fig. 1. Overview of the calibration architecture.

This paper is divided into seven sections. Section II gives a brief overview of the proposed calibration architecture, including its advantages and its limitations. Section III analyzes the errors in a 1.5-bit multiplying digital-to-analog converter (MDAC), then constructs the error model of an example two-stage pipelined ADC, and finally presents the overall error model of the whole pipelined ADC. Section IV describes the calibration details. In Section V, the prototype implementation is described. Measured results are given in Section VI, and finally a summary is presented in Section VII.

II. Calibration Architecture Overview

Fig. 1 shows an overview of the calibration architecture. It consists of an input sample-and-hold amplifier (SHA), a pipelined ADC core, a self-calibrated algorithmic ADC, and a digital post-processing block. The pipelined ADC is fast but not very accurate. It operates at a sampling rate of f_S. The algorithmic ADC is slow but accurate. It samples one out of every M SHA outputs V_{i1}, so it operates at sampling rate f_S/M. Corresponding pipelined and algorithmic outputs are compared in the digital post-processing block. The difference between these outputs is used to improve the linearity of the pipelined ADC. After calibration, the linearity of the pipeline is limited by the linearity of the algorithmic ADC.

In conventional algorithmic ADCs, the ideal residue gain is two, and deviations from this ideal gain limit the algorithmic ADC's linearity. Although analog techniques to reduce the residue gain error are known [8], [11], [15]–[19], they are

Manuscript received December 10, 2003; revised May 14, 2004. This work was supported by the National Science Foundation under Grant CCR-9901925 and by UC MICRO Grant 02-040.

X. Wang was with the University of California, Davis, CA 95616 USA. He is now with Marvell Semiconductor, Sunnyvale, CA 94086 USA.

P. J. Hurst and S. H. Lewis are with the Solid-State Circuits Research Laboratory, Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA (e-mail: lewis@ece.ucdavis.edu).

Digital Object Identifier 10.1109/JSSC.2004.835826

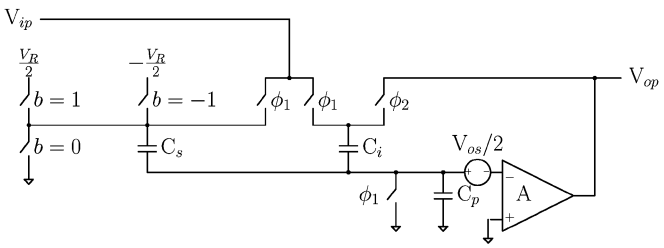


Fig. 2. Simplified half circuit of a 1.5-bit MDAC.

becoming more difficult to implement in modern CMOS technologies because process scaling is reducing available power supplies. To overcome this problem, the algorithmic ADC here is also calibrated in the digital domain. Since calibration is applied to the algorithmic ADC, which is then used to calibrate the pipelined ADC, the calibration here is nested. Furthermore, the algorithmic ADC need not be calibrated in the background because it is outside the main signal path. So calibrating the algorithmic ADC does not interrupt the flow of data through the pipelined ADC. Also, the algorithmic ADC can be small because it can be essentially just one stage of a pipelined ADC plus a SHA. Finally, the algorithmic ADC can have low power dissipation because it does not need to have low input-referred noise. High noise in the algorithmic ADC is allowed because the noise can be averaged out in the digital post-processing block as long as the goal is to track variations in residue gain that occur slowly (from temperature changes, for example).

III. Error Analysis

A. 1.5-Bit MDAC Error Analysis

The pipelined ADC uses a 1.5-bit/stage architecture. With this architecture, the main errors stem from errors in the residue gains provided by the MDACs. Fig. 2 shows the simplified half circuit of a fully differential 1.5-bit MDAC. A, V_{os}, and C_p denote the gain, offset voltage, and input parasitic capacitance of the opamp, respectively. In operation, the input V_{ip} is sampled onto C_s and C_i during phase ϕ_1. Then during phase ϕ_2, C_i is connected around the opamp, and C_s is connected to $\pm V_R/2$ or ground, depending on the raw digital output b.

Charge conservation analysis can be used to find the differential output V_{od} as

$$V_{od} = \frac{C_i + C_s}{\frac{C_{\rm tot}}{A} + C_i} V_{id} - \frac{C_s}{\frac{C_{\rm tot}}{A} + C_i} bV_R + \frac{C_{\rm tot}}{\frac{C_{\rm tot}}{A} + C_i} V_{os} \quad (1)$$

$$= 2(1+\epsilon_g)V_{id} - (1+\epsilon_{\rm DAC})bV_R + os' \quad (2)$$

where b is ± 1 or 0, V_{id} is the differential input, $C_{\rm tot} = C_i + C_s + C_p$, and os' is the output-referred amplifier offset voltage. If the opamp gain A is infinite, the input gain is $(C_i+C_s)/C_i$ because of feedforward from C_i, but the DAC gain is just C_s/C_i. Ideally, $C_s = C_i$. From (2), nonideality causes three errors in each MDAC: an input gain error ϵ_g, a DAC gain error $\epsilon_{\rm DAC}$, and offset os'. Note that the input gain error differs from the DAC gain error because feedforward increases the input gain but not the DAC gain.

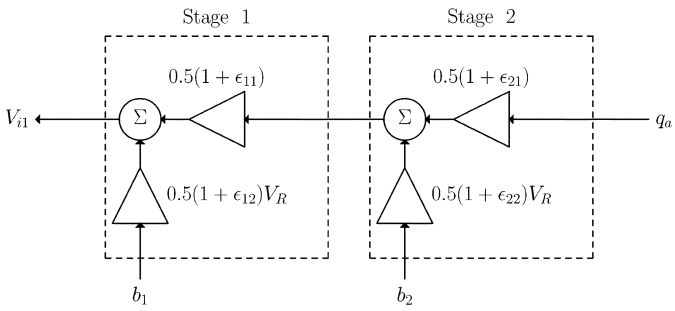


Fig. 3. Calculation of the input of a two-stage pipelined ADC for calibration. In stage k, b_k is the raw digital output; ϵ_{k1} is the output gain error, and ϵ_{k2} is the DAC gain error. The quantization error or residue of the last stage is q_a.

The stage output V_{od} is digitized by the rest of the pipeline. If the errors can be found, then the input corresponding to a given output can be calculated. So the calibration concept here is to compute the input from the digitized output and the errors. Based on this concept, the error model for the MDAC can be constructed by rewriting (2) and expressing the input V_{id} in terms of the output V_{od} and the errors.

B. Two-Stage Pipelined ADC Error Model

Fig. 3 shows the error model of an example two-stage pipelined ADC. It is similar to the model published in [4] except that it concentrates on calculating the input given the output instead of *vice versa*. For each stage, the input is calculated from the quantization error or residue of that stage, the raw digital output of that stage, and from the two gain errors of that stage. Offset error is ignored here because interstage offsets within the correction range can be combined into one equivalent input-referred offset in a 1.5-bit/stage ADC, and this total offset error can be handled separately. So for K stages, $2K$ gain error terms seem to be required for calibration. Calculation of the overall input V_{i1} in this two-stage example gives

$$V_{i1} = V_R[0.5(1+\epsilon_1)b_1 + 0.5^2(1+\epsilon_2)b_2] + q_a 0.5^2(1+\epsilon_3) \quad (3)$$

where $\epsilon_1 = \epsilon_{12}, \epsilon_2 = \epsilon_{11} + \epsilon_{22} + \epsilon_{11}\epsilon_{22}, \epsilon_3 = \epsilon_{11} + \epsilon_{21} + \epsilon_{11}\epsilon_{21}$, and q_a is the quantization error or residue of the last stage. Only three gain error terms are required: one for the raw output of each stage and another for the quantization error. So, for a two-stage pipeline, four parameters are needed in the error model: ϵ_1, ϵ_2, and ϵ_3 for gain errors and one more parameter for overall offset error. This result can be extended to any number of stages without any loss in generality. For a K-stage pipeline, $K+2$ parameters are needed in the error model (including one for the overall offset).

C. Pipelined ADC Error Model

Traditional pipelined ADCs have interstage gains of more than unity. Therefore, the significance of errors introduced by a given stage decreases as the stage under consideration moves down the pipeline. The prototype ADC in this project consists of 13 stages, but measured results described in Section VI show that only the first five stages contribute significant errors. Therefore, based on the analysis in Section III-B, $5+2$ or 7 parameters are required to characterize this pipeline.

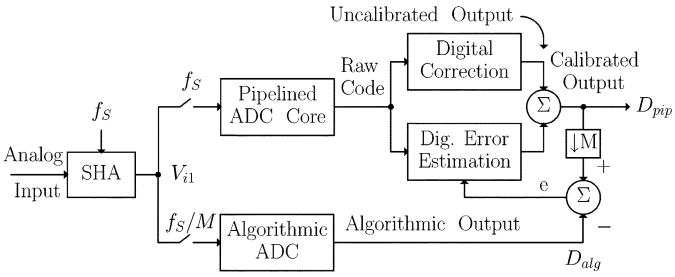


Fig. 4. Block diagram of pipelined ADC with calibration.

The normalized digitized input to the pipelined ADC with calibration $D_{\text{pip}}(V_{i1})$ can be expressed as

$$D_{\text{pip}}(V_{i1}) = \sum_{i=1}^{13} 0.5^i b_i + \sum_{i=1}^{5} 0.5^i \epsilon_i b_i + 0.5^5 \epsilon_6 q + os \quad (4)$$

where $q = \sum_{i=6}^{13} 0.5^{i-5} b_i$ is the digitized quantization error of the fifth stage and os is the overall ADC offset. $D_{\text{pip}}(V_{i1})$ consists of a sum of the weighted raw output codes of each stage $(b_1 \rightarrow b_{13})$ and then corrections from seven error terms. The first five of these are gain errors $(\epsilon_1 \rightarrow \epsilon_5)$ associated with the raw output codes of the first five stages $(b_1 \rightarrow b_5)$. The next is a gain error (ϵ_6) associated with the digitized quantization error q, and the last is the input-referred offset os. Although this error model has been applied to a 1.5-bit/stage pipeline in this work, it can be applied without modification for higher stage resolutions as well.

The key issue is how to find these seven error parameters. To find these parameters, a slow but accurate ADC is used to digitize some samples of the input voltage V_{i1}. The result is denoted by D_{alg}, which is the algorithmic ADC output for calibration. It can be subtracted from the calibrated digital output of the pipelined ADC (D_{pip}). The result is an error:

$$e = D_{\text{pip}} - D_{\text{alg}}. \quad (5)$$

The seven parameters can be chosen to minimize the mean-squared value of the error e using the least mean squared (LMS) algorithm.

IV. Background Calibration

A. Pipelined ADC and Its Calibration

Fig. 4 shows the detailed block diagram of the calibration architecture. The input SHA, the pipelined ADC core, and the digital correction block form a conventional pipelined ADC. Both the SHA and the pipelined ADC core operate at sampling rate f_S. The raw code of the pipelined ADC consists of two comparator outputs per stage and provides redundancy to overcome the effects of comparator offsets on the pipelined ADC linearity. This redundancy is removed in the digital correction block to produce the uncalibrated pipelined output. Furthermore, the raw code contains information about gain errors in the pipeline. This information is extracted in the digital error estimation block (DEE) and combined with the uncalibrated output to produce the calibrated output of the pipelined ADC. Meanwhile, the algorithmic ADC samples one out of every M SHA outputs

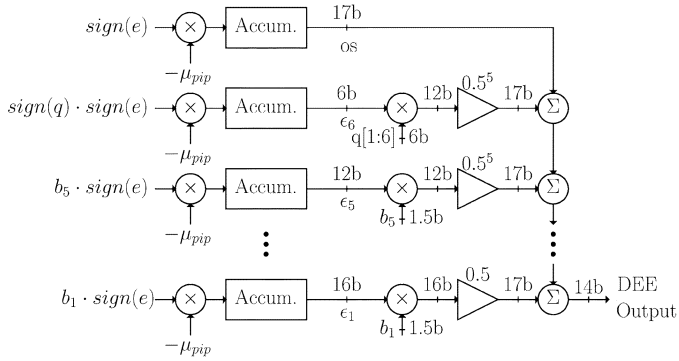


Fig. 5. Structure of the DEE (digital error estimation) block. $\text{sign}(e) = \pm 1$ and $q[1:6]$ is the first 6 bits of q, which is the digital code of the residue of the fifth stage q_a. All accumulators are 20 bits wide.

$(M = 32$ in the prototype). Corresponding pipelined and algorithmic outputs are subtracted, producing an error e. This error is used to update the seven parameters inside the DEE block to minimize the mean-squared error (MSE), $E(e^2)$, which is the the average value of e^2. Accumulators in the DEE block and negative feedback combine to minimize the MSE, so that the calibrated output approaches the algorithmic output in steady state.

The minimum MSE occurs where the derivative of the MSE with respect to each parameter is zero. For $\epsilon_1 \rightarrow \epsilon_5$, the condition is

$$\frac{\partial E(e^2)}{\partial \epsilon_i} = E\left(2e\frac{\partial e}{\partial \epsilon_i}\right) = E\left(2e\frac{\partial D_{\text{pip}}}{\partial \epsilon_i}\right) = 2(0.5^i)E\left(e \cdot b_i\right) = 0 \quad 1 \le i \le 5. \quad (6)$$

For ϵ_6

$$\frac{\partial E(e^2)}{\partial \epsilon_6} = E\left(2e\frac{\partial e}{\partial \epsilon_6}\right) = E\left(2e\frac{\partial D_{\text{pip}}}{\partial \epsilon_6}\right) = 2(0.5^5)E(e \cdot q) = 0. \quad (7)$$

For os

$$\frac{\partial E(e^2)}{\partial (os)} = E\left(2e\frac{\partial e}{\partial (os)}\right) = E\left(2e\frac{\partial D_{\text{pip}}}{\partial (os)}\right) = 2E(1 \cdot e) = 0. \quad (8)$$

These equations show that the condition for finding the minimum MSE is that the correlation between each signal [i.e., $b_1 \rightarrow b_5, q$ and a constant 1 that multiplies os in (4)] and the error e is zero.

Fig. 5 shows the structure of the DEE block. It contains seven accumulators. Each accumulator finds one parameter. For example, the bottom accumulator finds ϵ_1. ϵ_1 is the digital weighting applied to b_1, which is the raw output of the first pipeline stage. ϵ_1 is found here because the bottom accumulator operates on the product of b_1 and the sign of the error e. The sign of the error is used rather than the error itself to simplify the hardware without affecting the steady-state parameter values [20]. So with negative feedback, the average accumulator input is driven to zero, eliminating the correlation between b_1 and e. This happens only when ϵ_1 is found correctly, which means that the first stage output is properly weighted, and the part of e related to b_1 is zero. The other accumulators operate in a similar

way, each finding one of the other six parameters needed to calibrate the pipelined ADC. The digital outputs ($b_1 \rightarrow b_5$ and q) are weighted by six parameters ($\epsilon_1 \rightarrow \epsilon_6$) and added to the offset os to produce the DEE output.

The MSE is a second-order (quadratic) function of the unknown parameters; therefore, the MSE is a convex function of the parameters [20]. The adaptive loops adjust the ϵ_i's and os to minimize the MSE. A unique minimum will be found if the signals that are being weighted [i.e., $b_1 \rightarrow b_5, q$ and a constant 1 that multiplies os in (4)] are all different. On the other hand, if the sequences of bits b_1 and b_2 are identical for example, then the adaptive loops will not be able to determine unique coefficients for ϵ_1 and ϵ_2. In practice, a circuit could be added to detect whether the input signal is active enough to allow accurate calibration. This circuit would freeze the calibration when the input signal is not suitable for calibration (such as with a dc input signal for example).

The time required for the calibration of the pipelined ADC to converge depends on the value of $\mu_{\rm pip}$. With $\mu_{\rm pip} = 2^{-20}$, convergence accurate enough to give SNDR within 1 dB of its ideal value requires about one second. Increasing $\mu_{\rm pip}$ to 2^{-16} decreases the convergence time to about 60 ms but also reduces the peak SNDR by about 1 dB.

Many aspects of the hardware for the DEE block are simple. For example, the scaling factor $\mu_{\rm pip}$ is chosen as a power of 0.5; therefore, the multiplication by $\mu_{\rm pip}$ is implemented by simple bit shifting. Furthermore, the raw output codes ($b_1 \rightarrow b_5$) from the first five stages of the pipeline are ± 1 or 0, so the five multipliers that implement $\epsilon_i \times b_i, 1 \leq i \leq 5$, are simple. Finally, the multiplications in front of the bottom five accumulators are simple because they involve the product of a two-level or three-level code times a two-level code. Therefore, the only complicated multiplier is in the branch that finds ϵ_6. This multiplier computes $\epsilon_6 \times q[1:6]$ and requires 6-bit $\times$ 6-bit precision. In summary, the DEE block consists mainly of seven 20-bit accumulators, six 17-bit adders, and one 6-bit $\times$ 6-bit multiplier.

B. Algorithmic ADC and Its Calibration

Calibration of algorithmic ADCs has been demonstrated previously [12], [21], [22], and is simple because their performance depends primarily on one parameter: the residue gain. In [21], a trim array is used to adjust the residue gain to be as close to two as required. In [22], a digital calibration technique is described in which the ideal output of each stage during calibration was zero, allowing a residue amplifier stage with gain error to calibrate itself for errors introduced by capacitor mismatch. In [12], a digital calibration technique was described that could compensate for residue-gain errors introduced not only by capacitor mismatch, but also by finite opamp gain. The key was to examine the ADC output instead of the stage output and adjust the relative weighting of consecutive digital outputs until the major-carry jump is one LSB. The major-carry jump was measured by setting the ADC input to zero and finding the ADC output when the most significant bit (MSB) was set to one (D_1) and when the MSB was set to zero (D_0). Then $D_1 - D_0 = 1$ LSB under ideal conditions. An error $D_1 - D_0 - 1$ LSB was computed, and negative feedback adjusted the weighting of the comparator outputs to minimize the mean squared error.

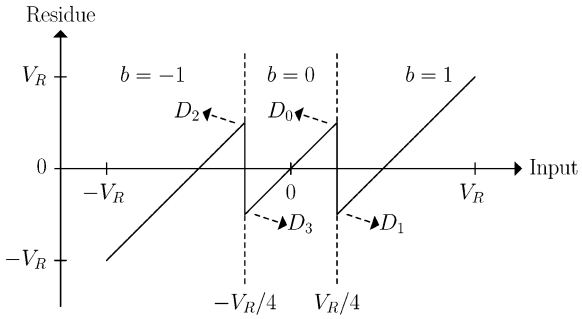


Fig. 6. Residue plot for the algorithmic ADC.

With this approach, a single comparator is used in the algorithmic ADC, and the residue amplifier gain is reduced so that nonidealities (such as nonzero offset in the comparator and residue amplifier gain error) do not produce residue outputs that saturate the remaining conversion range [23]. To increase the amount of comparator and residue amplifier offset that can be tolerated without reducing the residue gain, the algorithmic ADC described in this paper uses two comparators in a 1.5-bit/stage configuration [24]–[26]. Also, this approach allows the core of the algorithmic ADC to consist of one stage of the pipelined ADC plus a SHA, increasing modularity.

The digitized value of the input of the calibrated algorithmic ADC, D, can be expressed as a sum of the weighted raw outputs:

$$D = \sum_{i=1}^{N} m^i b_i \tag{9}$$

where $N = 16, m = 1/g, g$ is the equivalent residue gain of the algorithmic ADC, and b_i is the raw digital output of the ith comparison. Under ideal conditions, $m = 0.5$. Circuit nonidealities, such as capacitor mismatch and finite opamp gain, cause m to differ from its ideal value. So the key issue here is how to find the actual m.

Fig. 6 shows the plot of the residue of the algorithmic ADC versus its input. Ideally, the heights of the two residue jumps are identical. To find m, the algorithmic ADC input is set approximately equal to the upper comparator threshold ($V_R/4$), and the output is measured twice, yielding digital value D_1 when the corresponding comparator output is forced to 1 and D_0 when it is forced to 0. Then the process is repeated for the lower comparator threshold ($-V_R/4$), and the output is measured twice again, yielding D_2 when this comparator output is forced to 0 and D_3 when it is forced to 1.

With constant residue-gain error, two equations are established. That is, when m is found properly, $D_0 = D_1$, and $D_2 = D_3$. This result holds because the 1.5-bit/stage architecture provides redundancy. As a result, changing a comparator decision for an algorithmic ADC input near the threshold of that comparator should not change the digital output of the algorithmic ADC. Therefore, the calibration of the algorithmic ADC is based on calibration inputs of approximately $\pm V_R/4$. Its calibration does not depend on the properties of the input to be digitized by the pipelined ADC.

To compensate for constant gain errors, an error could be calculated and its mean squared value minimized. Two possible errors are

$$e = (D_0 - D_1) \tag{10}$$

and

$$e = (D_2 - D_3). \tag{11}$$

If the heights of the two residue jumps in Fig. 6 are equal, either error could be used. Consider the first error in (10). Its mean-squared value will be minimized when m is chosen so that

$$\frac{dE(e^2)}{dm} = E\left(2(D_0 - D_1)\frac{d(D_0 - D_1)}{dm}\right) = 0. \tag{12}$$

When $D_0 - D_1 = 0$, (12) is satisfied and the mean-squared value of the error in (10) will be minimized, as desired. If the estimate of m is found recursively, this estimate $\hat{m}$ can be calculated as

$$\hat{m}[j+1] = \hat{m}[j] - \mu_{\text{alg}} \cdot (D_0 - D_1) \tag{13}$$

where j is a time index.

In practice, some errors such as common-mode-to-differential-mode conversion in the residue amplifier can cause $D_0 - D_1$ to differ slightly from $D_2 - D_3$. Fortunately, such errors are relatively small in practice. To account for such differences, the squared error can be calculated as $(D_0 - D_1)^2 + (D_2 - D_3)^2$. Then the mean-squared error will be minimized when

$$\frac{dE(e^2)}{dm} = E\left(2(D_0 - D_1)\frac{d(D_0 - D_1)}{dm} + 2(D_2 - D_3)\frac{d(D_2 - D_3)}{dm}\right) = 0. \tag{14}$$

The estimate of m can be found using

$$\hat{m}[j+1] = \hat{m}[j] - \mu_{\text{alg}} \cdot \Delta D \tag{15}$$

where

$$\Delta D = (D_0 - D_1)\frac{d(D_0 - D_1)}{dm} + (D_2 - D_3)\frac{d(D_2 - D_3)}{dm}. \tag{16}$$

If $d(D_0 - D_1)/dm = d(D_2 - D_3)/dm$, ΔD can be simplified to

$$\Delta D = (D_0 - D_1) + (D_2 - D_3). \tag{17}$$

Although $d(D_0 - D_1)/dm$ is not equal to $d(D_2 - D_3)/dm$ under all conditions, they are both positive in the region under investigation and using (17) instead of (16) to calculate ΔD in the estimate of m for the algorithmic ADC has caused no more than 0.1 dB reduction in SNDR and SFDR of the pipelined ADC.

Fig. 7 shows the simplified model of the algorithmic ADC. It consists mainly of an input SHA followed by a 1.5-bit stage, which consists of a second SHA, two comparators, and a 1.5-bit DAC. The first and second SHAs have gains of g_1 and g_2, respectively. Therefore, the equivalent residue gain g is given by $g = g_1 \times g_2$. In operation, the algorithmic ADC rotates through

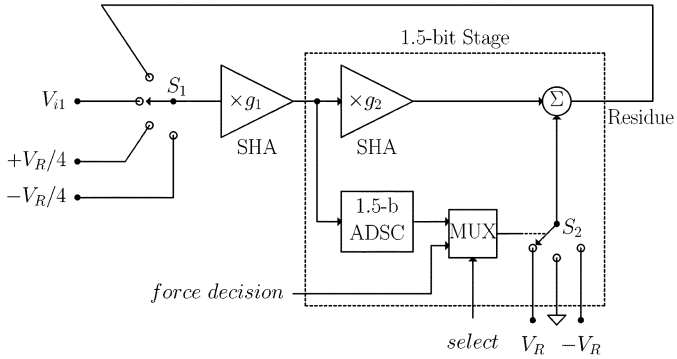


Fig. 7. Simplified model of the algorithmic ADC. ADSC stands for analog-to-digital subconverter.

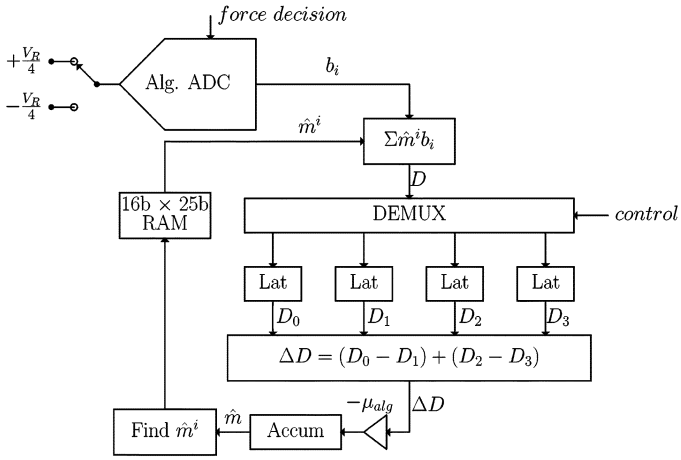


Fig. 8. Block diagram of the calibration loop for the algorithmic ADC.

the following sequence. First, it samples and digitizes the output of the SHA in Fig. 4 (V_{i1}). Then, it samples $V_R/4$ to find D_1. Then, it samples V_{i1} again followed by $V_R/4$ to find D_0. Next, it samples V_{i1} again followed by $-V_R/4$ to find D_3. Then, it samples V_{i1} again and finally $-V_R/4$ to find D_2. This sequence repeats indefinitely. Therefore, the algorithmic ADC alternates between digitizing the input V_{i1} and finding D_0, D_1, D_2, or D_3 for calibration. Two control signals (*force decision* and *select*) are introduced for calibration purposes.

Fig. 8 shows the block diagram of the calibration loop for the algorithmic ADC. During calibration, the algorithmic ADC samples one of its calibration inputs ($\pm V_R/4$) and produces a sequence of three-level codes $b_i, 1 \le i \le 16$. These codes are then scaled by their corresponding weighting factors $\hat{m}^i$ to construct the calibrated output code D according to (9). D is then steered by the demultiplexer into one of the latches, depending on which value of D has been measured.

After D_0, D_1, D_2, and D_3 have been measured, the error ΔD is found as given in (17). It is scaled by $-\mu_{\text{alg}}$ and added to the present estimate of m in the accumulator to produce the new estimate as in (15). This calibration loop has negative feedback and infinite dc gain because of the accumulator, so the average accumulator input, which is proportional to ΔD, is driven to zero when the proper value of m is found. The prototype uses

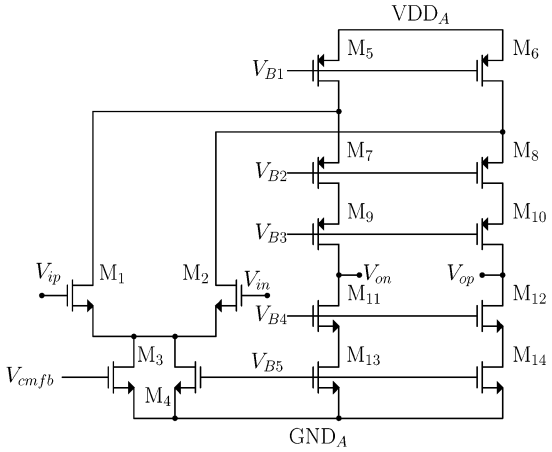


Fig. 9. Opamp schematic.

$\mu_{\rm alg} = 2^{-5}$, and calibration of the algorithmic ADC converges in about 0.2 ms.

To reduce the complexity of the digital logic, the powers of $\hat{m}$ can be calculated by a serial multiplier (labeled as "Find $\hat{m}^i$" in Fig. 8) and stored in a RAM [12]. All the arithmetic is done with 25-bit precision, except for the accumulator, which is 30 bits wide.

The algorithmic ADC uses an offset cancellation technique in which the signal polarity in the loop is reversed after the MSB decision is made [25]. Since the pipelined ADC is calibrated against an algorithmic ADC with low offset, the pipelined ADC offset with calibration is also low.

V. Prototype Implementation

Folded-cascode opamps [28] are used in both the pipelined and algorithmic ADCs. The schematic is shown in Fig. 9. Although the calibration described in this paper corrects for errors stemming from finite but constant opamp gain, it does not correct the nonlinearity caused by opamp gain variation. To overcome this problem, a pair of PMOS transistors (M_9 and M_{10}) were included to boost the simulated opamp gain from around 380 to around 2000. Although this opamp gain is large enough to make gain nonlinearity negligible, the remaining constant gain error limits the linearity of the uncalibrated pipelined ADC to below 9 bits. The voltage swing on each output of the opamp is from 1.0 to 1.8 V.

The input SHA shown in Fig. 10 is a flip-around structure that uses the folded-cascode opamp to provide high common-mode input range [29]. Compared with the non-flip-around SHA, it has lower kT/C noise, lower distortion introduced by opamp gain variation, and lower power dissipation due to higher feedback factor for a given speed requirement. Fig. 11 shows the fully differential comparator used in the first 3 stages of the pipelined ADC. It uses a switched-capacitor differencing circuit and provides some common-mode rejection [30]. Fig. 12 shows the comparator used in the last 10 stages of the pipeline and in the algorithmic ADC. The comparator core in both Figs. 11 and 12 uses the same schematic as in [31].

Note that the algorithmic ADC loads the input SHA in Fig. 4 only one out of every M cycles, where $M = 32$ in the prototype. This periodic loading might introduce interfering tones at f_S/M or 0.625 MHz. To avoid this problem, a dummy structure was added to make the static load on the input SHA constant for each sampled input. This dummy structure consists of a copy of the capacitor array, the switch array, and part of the opamp (with zero bias current) of the SHA of the algorithmic ADC, so that it resembles the input of the algorithmic ADC. This dummy loads the input SHA $M - 1$ out of every M cycles.

The sampling capacitance is scaled in several of the pipelined stages to reduce power dissipation [32], [33]. The sampling capacitance in the input SHA in Fig. 4 is 6 pF. For the next four SHAs in the pipelined ADC, the sampling capacitance is 2, 0.9, 0.4, and 0.2 pF, respectively. The rest of the stages in the pipeline use a sampling capacitance of 0.1 pF. The sampling capacitance in the algorithmic ADC is 0.2 pF. As a result, thermal noise limits its SNDR to about a 9- or 10-bit level. However, noise

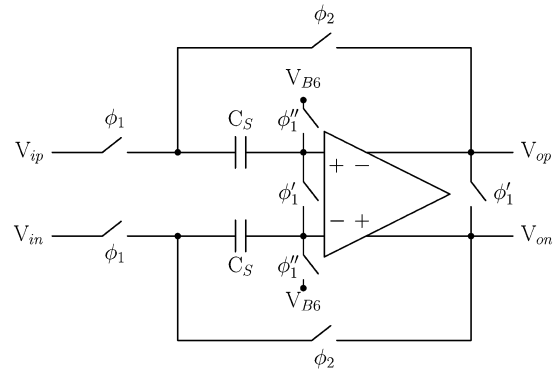


Fig. 10. Input sample-and-hold amplifier.

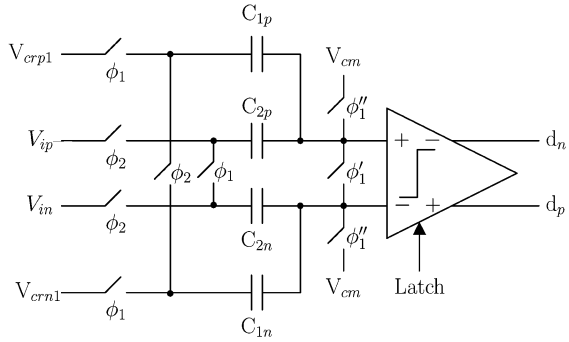


Fig. 11. Switched-capacitor comparator used in the first three stages of the pipeline ($C_{2p} = C_{2n} = 4C_{1p} = 4C_{1n} = 80$ fF).

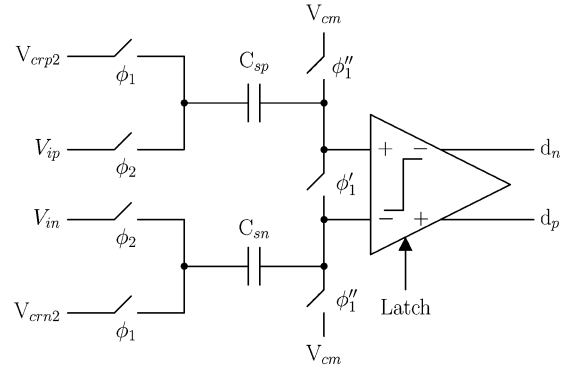


Fig. 12. Switched-capacitor comparator used in the last ten stages of the pipeline and in the algorithmic ADC ($C_{\rm sp} = C_{\rm sn} = 80$ fF).

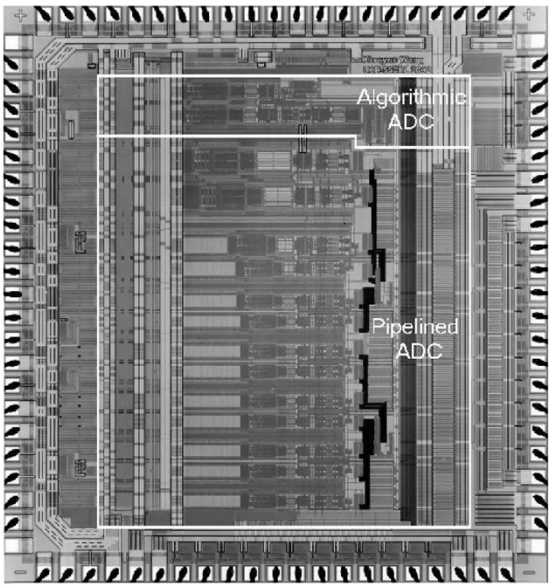


Fig. 13. Die photo.

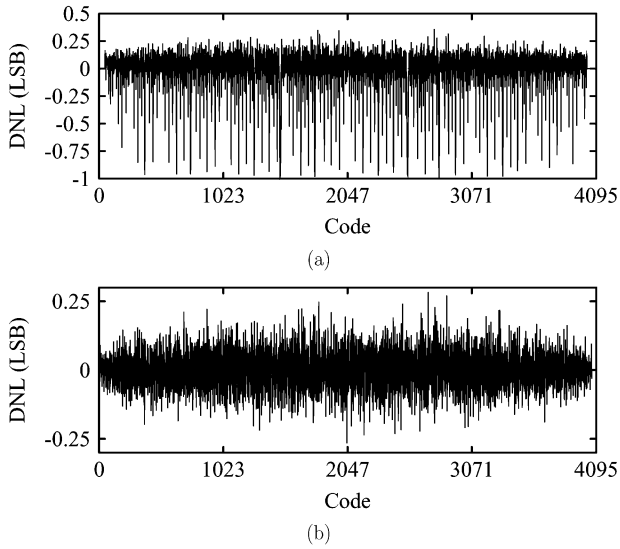


Fig. 14. Differential nonlinearity (DNL) of the algorithmic ADC: (a) without calibration and (b) with calibration. The input sample rate is 0.625 Msample/s.

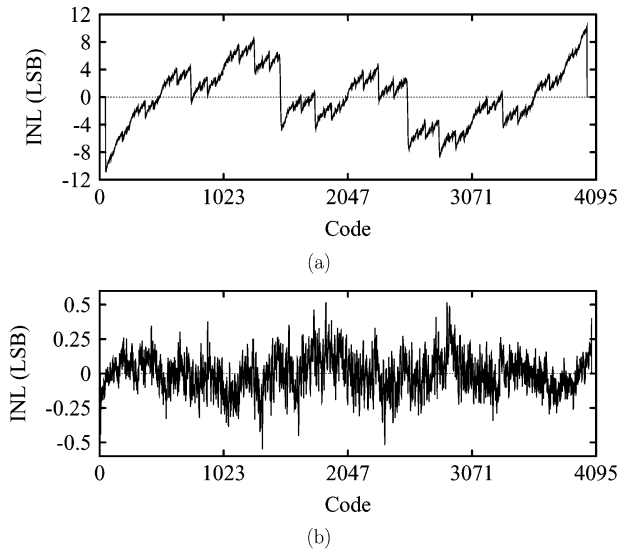


Fig. 15. Integral nonlinearity (INL) of the algorithmic ADC: (a) without calibration and (b) with calibration. The input sample rate is 0.625 Msample/s.

in the algorithmic ADC is not a limitation here because its effect is reduced by averaging in the accumulators in Fig. 5. To improve the power supply rejection ratio (PSRR) of the ADC system, NMOS supply bypassing capacitors were added extensively in the prototype.

Fig. 13 shows the die photo of the prototype. It is fabricated in a 0.35-μm CMOS technology and is approximately 3.5 mm $\times$ 3.8 mm. The total area is about 13.2 mm^2, and the active area is 7.5 mm^2. It consists of a 13-stage pipelined ADC and an algorithmic ADC. The digital calibration circuits are off-chip for simplicity. The required area of the digital calibration circuits is estimated to be around 1.7 mm^2, and their required power dissipation is estimated to be around 6 mW.

VI. Measured Results

Fig. 14 shows the differential nonlinearity (DNL) of the algorithmic ADC without and with calibration. The input frequency is 58 kHz, and the power supply is 3.3 V. These conditions remain the same for other figures unless stated otherwise. The sampling rate is 0.625 Msample/s. Without calibration, the peak DNL is about -1 LSB, and many codes are missing. With calibration, the peak DNL decreases to 0.28 LSB.

Fig. 15 shows the INL of the algorithmic ADC without and with calibration. Without calibration, the peak INL is 10.9 LSB. With calibration, the peak INL decreases to -0.55 LSB. The ultimate linearity is mainly limited by opamp gain variation. This effect is worse for the algorithmic ADC than for the pipelined ADC because the residue in the algorithmic ADC is processed by two opamps between raw bit decisions.

Fig. 16 shows the DNL of the pipelined ADC without and with calibration. The sampling rate is 20 Msample/s. Without calibration, the peak DNL is -0.60 LSB. With calibration, the peak DNL decreases to -0.41 LSB.

Fig. 17 shows the INL of the pipelined ADC without and with calibration. Without calibration, the peak INL is -4.21 LSB. With calibration, the peak INL decreases to -0.47 LSB. Note that, with calibration, the peak INL of the pipelined ADC is better than that of the algorithmic ADC. This is because the calibration here is based on a statistical identification of the seven parameters. In contrast, an alternative calibration procedure would store the differences of all 2^{12} corresponding pipelined and algorithmic ADC output codes. With this alternative procedure, the INL of the pipelined ADC would exactly follow the INL pattern of the algorithmic ADC. For the calibration method used here, however, the INL of the pipelined ADC is not forced to equal the INL of the algorithmic ADC on a code-by-code basis. Therefore, the peak INL of the pipeline need not be the same as that of the algorithmic ADC.

Fig. 18 shows the output spectra of the pipelined ADC without and with calibration. The input magnitude is 1.6 Vp-p. The sampling rate is 20 Msample/s, but the pipelined ADC output was downsampled by a factor of four on the chip to reduce the required number of output pins. Therefore, these

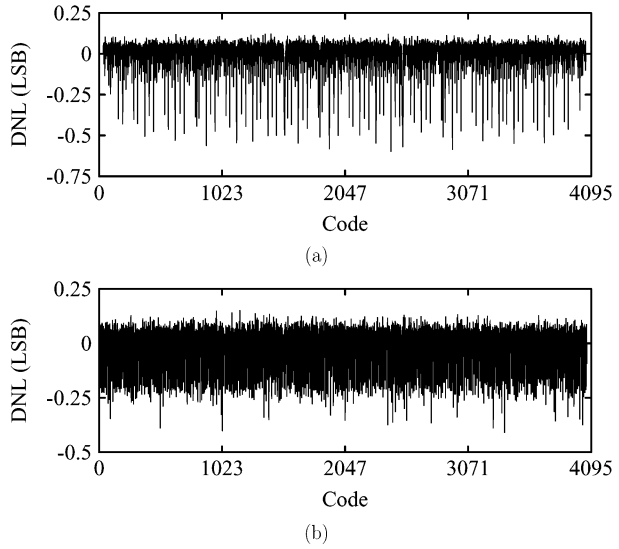


Fig. 16. Differential nonlinearity (DNL) of the pipelined ADC: (a) without calibration and (b) with calibration. The sample rate is 20 Msample/s.

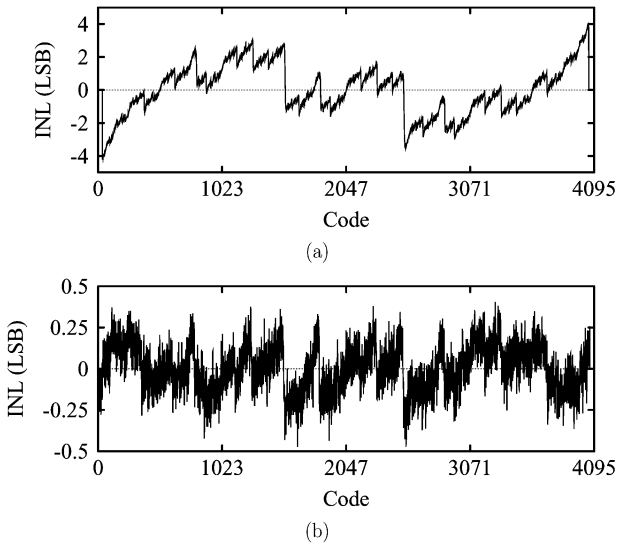


Fig. 17. Integral nonlinearity (INL) of the pipelined ADC: (a) without calibration and (b) with calibration. The sample rate is 20 Msample/s.

two plots extend to half the sample rate divided by four, or 2.5 MHz. Fig. 18(a) shows the output spectra of the ADC without calibration. Without calibration, the SNDR is 58.2 dB, which corresponds to about 9.4 effective bits, and the SFDR is 59.4 dB. Fig. 18(b) shows the output spectra of the ADC with calibration. Notice that the distortion tones introduced by circuit nonidealities have been successfully removed. Also note that the tone at the algorithmic ADC sample rate of 0.625 MHz is 107.5 dB below the input, so the interference from the loading of the algorithmic ADC is not significant here. The overall SNDR of the pipelined ADC has improved to 70.8 dB, which corresponds to about 11.5 effective bits, and the SFDR has improved to 93.3 dB. The THD is -92.9 dB with a dominant fifth harmonic of -98.6 dB. The other harmonics are all below -100 dB. Although some recent ADCs [29], [34], [35] have used stage resolutions greater than 1.5 bits in the first stage to improve SFDR, the SFDR and THD reported in this paper are

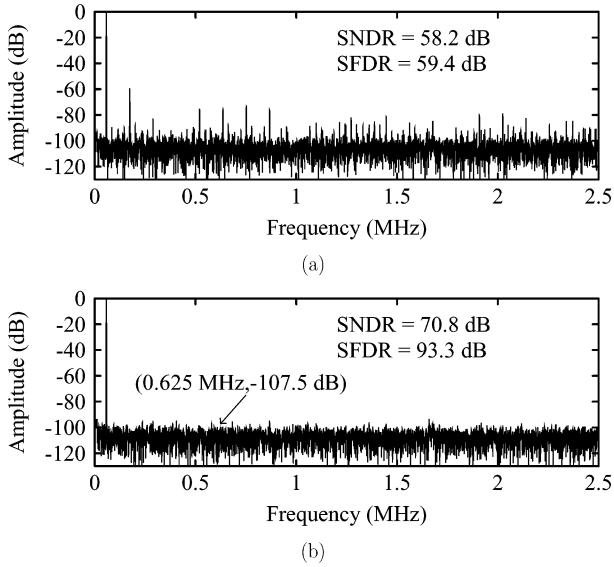


Fig. 18. Pipelined ADC output spectra for $f_S = 20$ Msample/s (downsampled by a factor of 4), $V_{\rm in} = 1.6$ Vp-p and $f_{\rm in} = 58$ kHz: (a) without calibration and (b) with calibration.

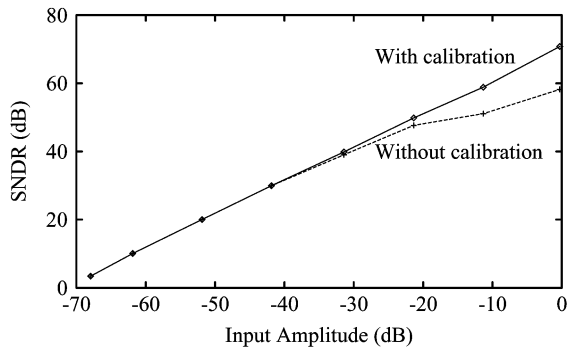


Fig. 19. SNDR versus input amplitude for the pipelined ADC without and with calibration, $f_S = 20$ Msample/s.

the highest reported for CMOS ADCs with sample rates of at least 20 Msample/s.

Fig. 19 shows two plots of SNDR versus input amplitude. The dashed plot is without calibration, and the solid plot is with calibration. For small input amplitude, these two plots merge because the raw output codes from the first five stages are all zero for small inputs, so calibration does not affect the result in this case. For large input amplitude, however, calibration improves the maximum SNDR from 58.2 to 70.8 dB.

Fig. 20(a) shows two plots of SNDR versus input frequency. The dashed plot is without calibration, and the solid plot is with calibration. The SNDR decreases with increasing input frequency. With calibration, the SNDR is above 68.9 dB for input frequencies up to 3 MHz. This limit stems from sampling jitter, which is about 12 ps rms.

Fig. 20(b) shows two plots of SFDR versus input frequency. The dashed plot is without calibration, and the solid plot is with calibration. With calibration, the SFDR is above 87.7 dB for input frequencies up to 3 MHz.

Fig. 21 shows plots of SNDR and SFDR versus the number of stages that are calibrated ($N_{\rm cal}$). $N_{\rm cal}$ varies from 0 to 10, and 0 corresponds to the case without calibration. The SNDR and

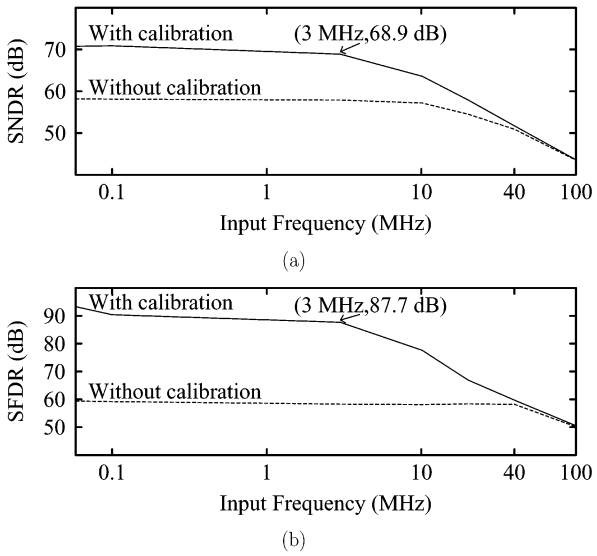


Fig. 20. (a) SNDR and (b) SFDR versus input frequency for the pipelined ADC without and with calibration, $f_S = 20$ Msample/s.

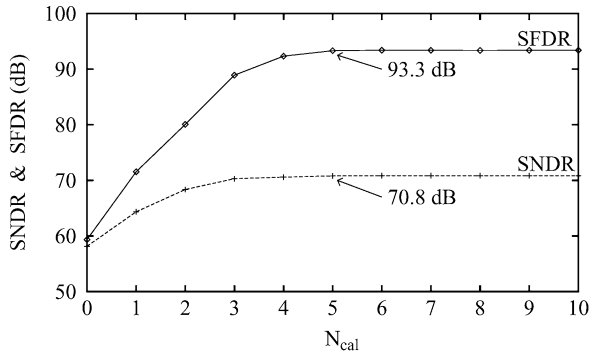


Fig. 21. SNDR and SFDR of the pipelined ADC versus the number of calibrated stages.

TABLE I
PERFORMANCE SUMMARY (3.3 V, 25 °C)

Process	0.35μm 2P4M CMOS	
Sampling rate	20 Msample/s	
Active area	7.5 mm^2	
Full-Scale Input	1.6 Vp-p	
	Without Cal.	With Cal.
Analog Power Diss.	190 mW	226 mW
Total Power Diss.	217 mW	254 mW
Max. |INL| (Pip. ADC)*	4.21 LSB	0.47 LSB
Max. |DNL| (Pip. ADC)*	0.60 LSB	0.41 LSB
SNDR (Alg. ADC)*	49.6 dB	59.6 dB
SNDR (Pip. ADC)*	58.2 dB	70.8 dB
SFDR (Pip. ADC)*	59.4 dB	93.3 dB
THD (Pip. ADC) *	–59.4 dB	–92.9 dB
PSRR*	65.0 dB	64.8 dB
CMRR*	73.6 dB	73.4 dB

*f_{in}=58 kHz

SFDR increase when the number of stages that are calibrated increase until N_{cal} reaches five. After that, the performance is about constant, which explains why five stages are calibrated for the data shown in this paper unless stated otherwise. Table I summarizes the performance of the prototype.

VII. CONCLUSION

Speed–accuracy tradeoffs are natural in ADCs, so high-speed ADCs are not normally as accurate as low-speed ADCs. This paper has described a new ADC architecture that uses a slow but accurate ADC to calibrate a fast but inaccurate ADC. Furthermore, the slow but accurate ADC is itself calibrated, which means that the calibration is nested. The result is a fast and accurate ADC that does not require any high-gain opamps.

REFERENCES

[1] T.-H. Shu, B.-S. Song, and K. Bacrania, "A 13-b 10-Msample/s ADC digitally calibrated with oversampling delta-sigma converter," *IEEE J. Solid-State Circuits*, vol. 30, pp. 443–452, Apr. 1995.

[2] I. Galton, "Digital cancellation of D/A converter noise in pipelined A/D converters," *IEEE Trans. Circuits Syst. II*, vol. 47, pp. 185–196, Mar. 2000.

[3] P. C. Yu, S. Shehata, A. Joharapurkar, P. Chuch, A. R. Bugeja, X. Du, S.-U. Kwak, Y. Papantonopoulous, and T. Kuyel, "A 14b 40 Msample/s pipelined ADC with DFCA," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2001, pp. 136–137.

[4] U.-K. Moon and B.-S. Song, "Background digital calibration techniques for pipelined ADC's," *IEEE Trans. Circuits Syst. II*, vol. 44, pp. 102–109, Feb. 1997.

[5] S.-U. Kwak, B.-S. Song, and K. Bacrania, "A 15-b 5-Msample/s low-spurious CMOS ADC," *IEEE J. Solid-State Circuits*, vol. 32, pp. 1866–1875, Dec. 1997.

[6] R. Jewett, K. Poulton, K.-C. Hsieh, and J. Doernberg, "A 12-b 128-Msample/s ADC with 0.05 LSB DNL," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 1997, pp. 138–139.

[7] D. Fu, K. C. Dyer, S. H. Lewis, and P. J. Hurst, "A digital background calibration technique for time-interleaved analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1904–1911, Dec. 1998.

[8] J. Ming and S. H. Lewis, "An 8-bit 80-Msample/s pipelined analog-to-digital converter with background calibration," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1489–1497, Oct. 2001.

[9] E. J. Siragusa and I. Galton, "Gain error correction technique for pipelined analogue-to-digital converters," *Electron. Lett.*, pp. 617–618, Mar. 2000.

[10] B. Murmann and B. E. Boser, "A 12-b 75-MS/s pipelined ADC using open-loop residue amplification," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2003, pp. 328–329.

[11] J. M. Ingino and B. A. Wooley, "A continuously calibrated 12-b, 10-MS/s, 3.3-V A/D converter," *IEEE J. Solid-State Circuits*, vol. 33, pp. 1920–1931, Dec. 1998.

[12] O. E. Erdoğan, P. J. Hurst, and S. H. Lewis, "A 12-b digital-background-calibrated algorithmic ADC with -90-dB THD," *IEEE J. Solid-State Circuits*, vol. 34, pp. 1812–1820, Dec. 1999.

[13] E. B. Blecker, T. M. McDonald, O. E. Erdoğan, P. J. Hurst, and S. H. Lewis, "Digital background calibration of an algorithmic analog-to-digital converter using a simplified queue," *IEEE J. Solid-State Circuits*, vol. 38, pp. 1059–1062, June 2003.

[14] X. Wang, P. J. Hurst, and S. H. Lewis, "A 12-bit 20-MS/s pipelined ADC with nested digital background calibration," in *Proc. IEEE Custom Integrated Circuits Conf.*, Sept. 2003, pp. 409–412.

[15] P. W. Li, M. J. Chin, P. R. Gray, and R. Castello, "A ratio-independent algorithmic analog-to-digital conversion technique," *IEEE J. Solid-State Circuits*, vol. SC-19, pp. 828–836, Dec. 1984.

[16] C. Shih and P. R. Gray, "Reference refreshing cyclic analog-to-digital and digital-to-analog converters," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 544–554, Aug. 1986.

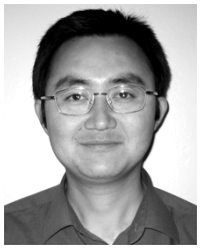
[17] S. Sutarja and P. R. Gray, "A pipelined 13-bit 250-ks/s 5-V analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1316–1323, Dec. 1988.

[18] B.-S. Song, M. F. Tompsett, and K. R. Lakshmikumar, "A 12-bit 1-Msample/s capacitor error-averaging pipelined A/D converter," *IEEE J. Solid-State Circuits*, vol. SC-23, pp. 1324–1333, Dec. 1988.

[19] Y.-M. Lin, B. Kim, and P. R. Gray, "A 13-b 2.5-MHz self-calibrated pipelined A/D converter in 3-μm CMOS," *IEEE J. Solid-State Circuits*, vol. 26, pp. 628–636, Apr. 1991.

[20] M. L. Honig and D. G. Messerschmitt, *Adaptive Filters: Structures, Algorithms, and Applications*. Boston, MA: Kluwer, 1984.

[21] H. Ohara, H. X. Ngo, M. J. Armstrong, C. F. Rahim, and P. R. Gray, "A CMOS programmable self-calibrating 13-bit eight-channel data acquisition peripheral," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 930–938, Dec. 1987.
[22] H. S. Lee, "A 12-b 600 ks/s digitally self-calibrated pipelined algorithmic ADC," *IEEE J. Solid-State Circuits*, vol. 29, pp. 509–515, Apr. 1994.
[23] A. Karanicolas, H.-S. Lee, and K. Bacrania, "A 15-b 1 Msample/s digitally self-calibrated pipeline ADC," *IEEE J. Solid-State Circuits*, vol. 28, pp. 1207–1215, Dec. 1993.
[24] G. Jusuf, "A 1-bit/cycle algorithmic analog-to-digital converter without high-precision comparators,", Univ. California, Berkeley, Memorandum UCB/ERL M90/69, Aug. 1990.
[25] B. Ginetti, P. G. A. Jespers, and A. Vandemeulebroecke, "A CMOS 13-b cyclic RSD A/D converter," *IEEE J. Solid-State Circuits*, vol. 27, pp. 957–965, July 1992.
[26] K. Nagaraj, "Efficient circuit configurations for algorithmic analog to digital converters," *IEEE Trans. Circuits Syst. II*, vol. 40, pp. 777–785, Dec. 1993.
[27] O. E. Erdoğan, "Digital background calibration of analog-to-digital converters using a calibration queue," Ph.D. dissertation, Univ. of California, Davis, 1999.
[28] T. C. Choi, R. T. Kaneshiro, R. W. Brodersen, P. R. Gray, W. B. Jett, and M. Wilcox, "High-frequency CMOS switched-capacitor filters for communication applications," *IEEE J. Solid-State Circuits*, vol. SC-18, pp. 652–664, Dec. 1983.
[29] W. Yang, D. Kelly, I. Mehr, M. T. Sayuk, and L. Singer, "A 3-V 340-mW 14-b 75-Msample/s CMOS ADC with 85-dB SFDR at Nyquist input," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1931–1936, Dec. 2001.
[30] T. Shih, L. Der, S. H. Lewis, and P. J. Hurst, "A fully differential comparator using a switched-capacitor differencing circuit with common-mode rejection," *IEEE J. Solid-State Circuits*, vol. 32, pp. 250–253, Feb. 1997.
[31] K. Nagaraj, H. S. Fetterman, J. Anidjar, S. H. Lewis, and R. G. Renninger, "A 250-mW, 8b, 52-Msamples/s parallel-pipelined A/D converter with reduced number of amplifiers," *IEEE J. Solid-State Circuits*, vol. 32, pp. 312–320, Mar. 1997.
[32] T. B. Cho and P. R. Gray, "A 10 b, 20 Msample/s, 35 mW pipeline A/D converter," *IEEE J. Solid-State Circuits*, vol. 30, pp. 166–172, Mar. 1995.
[33] D. W. Cline and P. R. Gray, "A power optimized 13-b 5 Msamples/s pipelined analog-to-digital converter in 1.2 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 31, pp. 294–303, Mar. 1996.
[34] L. Singer, S. Ho, M. Timko, and D. Kelly, "A 12-b 65 Msample/s CMOS ADC with 82 dB SFDR at 120 MHz," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2000, pp. 38–39.
[35] H. Pan, M. Segami, M. Choi, J. Cao, and A. A. Abidi, "A 3.3-V 12-b 50-MS/s A/D converter in 0.6-μm CMOS with over 80-dB SFDR," *IEEE J. Solid-State Circuits*, vol. 35, pp. 1769–1780, Dec. 2000.



Xiaoyue Wang (S'99–M'04) was born in Yunnan, China, in 1972. He received the B.S. and M.S. degrees from Fudan University, China, in 1995 and 1998, respectively, and the Ph.D. degree in electrical and computer engineering from the University of California at Davis in 2003.

He is currently a Staff Design Engineer with Marvell Semiconductor, Sunnyvale, CA. His research interests include design of integrated circuits and systems for wireless communications.

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 at Berkeley in 1977, 1979, and 1983, respectively.

From 1983 to 1984, he was with the University of California at Berkeley, as a Lecturer, teaching integrated circuit design courses and working on an MOS delta-sigma modulator. In 1984, he joined the telecommunications design group of Silicon Systems Inc., Nevada City, CA, 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 coauthor 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 was a guest editor for the December 1999 issue of the JOURNAL OF SOLID-STATE CIRCUITS, and is now an Associate Editor for the JOURNAL.

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

From 1980 to 1982, he was with Bell Laboratories, Whippany, NJ, where he was involved in circuit design for magnetic recording. In 1988, he rejoined Bell Laboratories, Reading, PA, 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 at Davis, where he is now a Professor. He is a coauthor of a college textbook on analog integrated circuits. His research interests include data conversion, signal processing, and analog circuit design.

Dr. Lewis received the award for the Outstanding Engineering Scholar at Rutgers University, the Sakrison Memorial Prize at the University of California at Berkeley, and the IEEE Third Millennium Medal. He was also a corecipient of the Jack Kilby Award for Outstanding Student Paper and the Beatrice Winner Award for Editorial Excellence at ISSCC. He was a member of the Program Committee for the International Solid-State Circuits Conference from 1994 to 1998, an Associate Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS from 1994 to 1997, and Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS from 1998 to 2001. He is now President of the IEEE Solid-State Circuits Society.