Digital Background Correction of Harmonic Distortion in Pipelined ADCs

Andrea Panigada, *Student Member, IEEE*, and Ian Galton, *Member, IEEE*

***Abstract*—Pipelined analog-to-digital converters (ADCs) are sensitive to distortion introduced by the residue amplifiers in their first few stages. Unfortunately, residue amplifier distortion tends to be inversely related to power consumption in practice, so the residue amplifiers usually are the dominant consumers of power in high-resolution pipelined ADCs. This paper presents a background calibration technique that digitally measures and cancels ADC error arising from distortion introduced by the residue amplifiers. It allows the use of higher distortion and, therefore, lower power residue amplifiers in high-accuracy pipelined ADCs, thereby significantly reducing overall power consumption relative to conventional pipelined ADCs.**

***Index Terms*—Analog-to-digital conversion, calibration, harmonic distortion, mixed analog–digital integrated circuits (ICs).**

I. Introduction

PIPELINED analog-to-digital converters (ADCs) are widely used in applications that require data converters with resolutions in the range of 10 to 16 bits and bandwidths in the range of 15 to 250 MHz [1]–[14]. Such applications include cellular telephone base station receivers, 802.11 wireless LAN receivers, and 802.16 wireless metropolitan area network receivers. In general, pipelined ADCs are attractive when the required bandwidth is too high for oversampling delta–sigma ADCs to be efficient and the required resolution is too high for flash ADCs to be efficient.

Unfortunately, the power consumption of high-resolution pipelined ADCs tends to be large, mainly because of the high-performance op-amps required in the first few pipeline stages. Passive sampling can be used to avoid having an op-amp based sample-and-hold in the first stage, which leaves the op-amps in the residue amplifiers of the first few stages as the dominant consumers of power [15], [16]. Each stage in a pipelined ADC performs coarse digitization of its input signal, but the outputs of the stages are combined such that most of the quantization noise cancels to achieve a high-resolution digitized version of the input signal. However, distortion introduced by the residue amplifiers, particularly those in the first few stages, results in imperfect cancellation which reduces the linearity of the pipelined ADC and increases its noise floor. In general, high op-amp gain and bandwidth are required to achieve sufficiently low-distortion closed-loop residue amplifier performance. If it were not for this limitation, much lower performance op-amps could be used in pipelined ADCs to significantly reduce power consumption.

This paper presents a digital background calibration technique, called the *harmonic distortion correction (HDC) technique*, that digitally measures and cancels ADC error arising from distortion introduced by the residue amplifiers. This makes it possible to reduce the power consumption of the op-amps in a given pipelined ADC without sacrificing ADC accuracy. The HDC technique operates in background during normal operation of the pipelined ADC, so it adapts to environmental changes without the need to interrupt normal operation of the ADC. As with other digital calibration techniques, such as presented in [17] and [18], the HDC technique requires a significant amount of digital signal processing. However, the reduction in op-amp power consumption is expected to far exceed the increase in power consumption from the extra digital logic.

The HDC technique is based on a different principle of operation than the only other techniques known to the authors that cancel harmonic distortion in pipelined ADCs [19], [20]. The benefit of the HDC technique relative to that presented in [19] is that it works for any pipelined ADC input signal, and the benefits relative to that presented in [20] are that it does not have restrictions with respect to dc input signals and it is not sensitive to amplifier offsets.

The paper consists of three main sections. Section II presents an example pipelined ADC architecture and describes the residue amplifier distortion problem. Section III presents the signal processing details underlying the HDC technique. Section IV presents an implementation example of the HDC technique. Section V presents simulation results and limitations of the HDC technique.

II. Residue Amplifier Distortion Problem

A. Example Pipelined ADC

A seven-stage example pipelined ADC architecture is shown in Fig. 1. Each stage except the last consists of a 9-level flash ADC, a 9-level digital-to-analog converter (DAC), and a *residue amplifier* with a gain of 4. The last stage consists of just a 9-level flash ADC. All the flash ADCs and DACs are clocked simultaneously at a sample rate of $f_s = 1/T_s$. The ideal behavior of each flash ADC is to update its digital output each sample time to whichever of the 9 values, $-4\Delta, -3\Delta, \ldots, 4\Delta$, is closest to the input voltage at that sample time, where Δ is the quantization *step size* of the flash ADC. Therefore, from a signal processing point of view, each flash ADC ideally acts as a 9-level

Manuscript received November 1, 2005; revised April 1, 2006. This work was supported in part by STMicroelectronics and in part by the University of California Discovery Program and various corporations under Grant com04-10177. This paper was recommended by Associate Editor G. Palumbo.

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

Digital Object Identifier 10.1109/TCSI.2006.880034

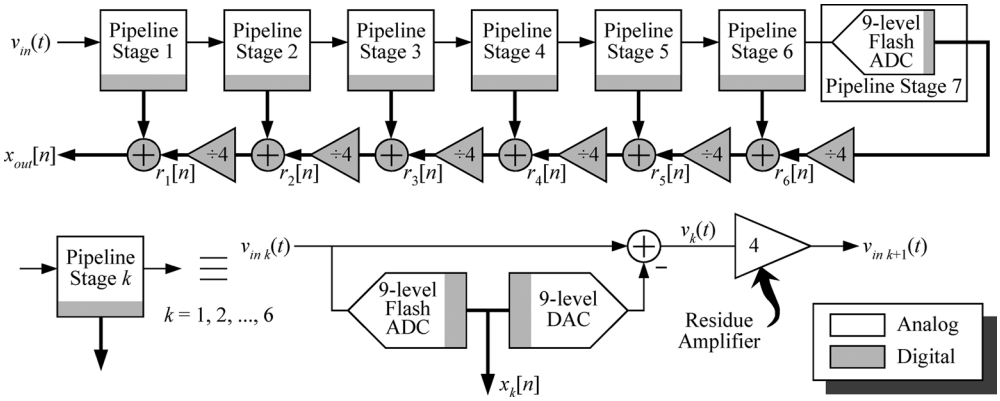

Fig. 1. Block diagram of an example 15-bit pipelined ADC.

uniform quantizer, and the output of the kth flash ADC is given by

$$x_k[n] = v_{\text{in}k}(nT_s) + e_{\text{ADC}k}[n] \tag{1}$$

where $v_{\text{in}k}(t)$ is the flash ADC's input signal, and $e_{\text{ADC}k}[n]$ is the *quantization error* introduced by the flash ADC. The input *no-overload range* of each flash ADC, and, therefore, the *usable input range* of each pipeline stage, is -4.5Δ to 4.5Δ, because the magnitude of the *quantization error* introduced by the flash ADC is bounded by $\Delta/2$ for input voltages within this range and exceeds $\Delta/2$ otherwise. The ideal behavior of each DAC is to convert the format of its input from a digital representation (e.g., bits) to an analog representation (e.g., voltage) without introducing distortion or noise. Therefore, from a signal processing point of view an ideal DAC performs no numerical operation. It follows that in the absence of nonideal circuit behavior the input to and output of the kth residue amplifier at the nth sample time are given, respectively, by

$$v_k(nT_s) = -e_{\text{ADC}k}[n], \text{ and } v_{\text{in}k+1}(nT_s) = 4v_k(nT_s). \tag{2}$$

The outputs of the flash ADCs are combined as shown in Fig. 1 to form the output of the pipelined ADC, $x_{\text{out}}[n]$. The output, $r_k[n]$, of each digital divide-by-four block is called the *digitized residue* of the kth stage. As can be seen from Fig. 1, $r_k[n] = (r_{k+1}[n] + x_{k+1}[n])/4$ for $k = 1, 2, \ldots, 5$, so recursive application of (1) and (2) gives

$$r_k[n] = v_k(nT_s) + \frac{1}{4^{7-k}} e_{\text{ADC}7}[n]. \tag{3}$$

Hence, in the absence of nonideal circuit behavior the quantization error sequences from all but the last pipeline stage cancel to give

$$x_{\text{out}}[n] = v_{\text{in}}(nT_s) + \frac{1}{4096} e_{\text{ADC}7}[n]. \tag{4}$$

Since $e_{\text{ADC}7}[n]$ is bounded in magnitude by $\Delta/2$ and the first pipeline stage has a usable input range of -4.5Δ to 4.5Δ, this represents slightly more than 15-bit analog-to-digital conversion accuracy.

With ideal circuit behavior, the magnitude of the quantization error from each flash ADC is bounded by $\Delta/2$, so the analog output of each pipeline stage ideally never exceeds 2Δ in magnitude. However, nonideal circuit behavior such as comparator offset voltages can cause the analog outputs of the pipeline stages to have magnitudes that exceed 2Δ from time to time. To accommodate such *over-range* conditions, the useable input range of the second through seventh pipelined stages is maintained at -4.5Δ to 4.5Δ instead of -2Δ to 2Δ. In this case, the pipelined ADC is said to have an *over-range* margin of $\pm 2.5\Delta$. The over-range margin greatly relaxes the performance requirements of the flash ADCs in pipelined ADCs [21].

B. Effect of Residue Amplifier Distortion

The effect of residue amplifier distortion can be demonstrated by considering the pipelined ADC of Fig. 1 with all ideal components except for the residue amplifier in the first stage. This scenario is shown in Fig. 2, wherein a function, f, is used to represent distortion introduced by the first stage's residue amplifier.

The distortion introduced by a practical residue amplifier tends to be well-modeled as a memoryless, weakly nonlinear function of the amplifier's input voltage, so it can be approximated accurately by its first N Taylor series coefficients where N typically is small (e.g., $N \leq 5$ is common). Consequently, the distortion function, f, in Fig. 2 is given by

$$f(v_1) = \sum_{n=1}^{N} \alpha_n v_1^n. \tag{5}$$

The same argument used above to obtain (4) implies that the output of the pipelined ADC is now

$$x_{\text{out}}[n] = x_{\text{out}}[n]|_{\text{ideal}} + f\left(v_1(nT_s)\right) \tag{6}$$

where $x_{\text{out}}[n]|_{\text{ideal}}$ is the ideal output of the pipelined ADC given by (4).

For example, suppose $\alpha_n = 0$ for all n except $n = 1$. This implies that the distortion is just a gain error, i.e., linear distortion. In the absence of other nonideal circuit behavior, $v_1(nT_s) = e_{\text{ADC}1}[n]$ and is, therefore, bounded in magnitude by $\Delta/2$, so it follows from (6) that the maximum magnitude of the error from

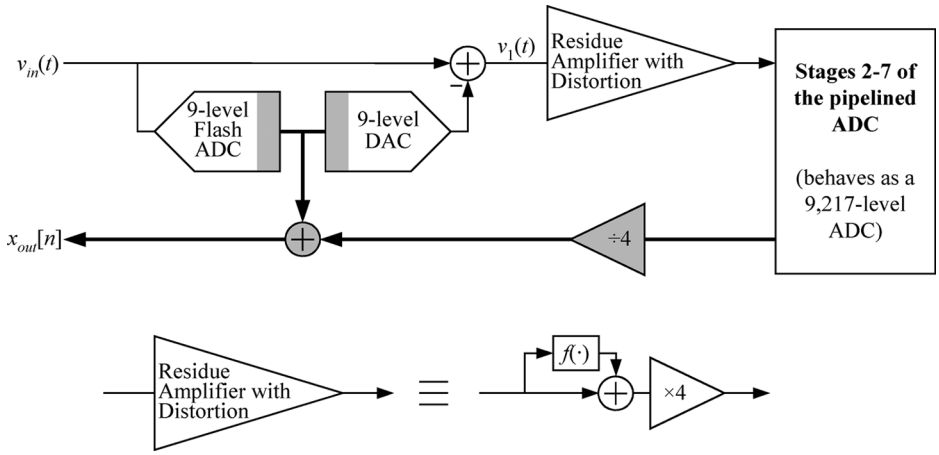

Fig. 2. Pipelined ADC of Fig. 1 except with a residue amplifier in the first stage that introduces distortion.

the nonideal residue amplifier gain is $f(\Delta/2) = |\alpha_1|\Delta/2$. It follows from (4) that the quantization error introduced by the ideal version of the pipelined ADC has a maximum magnitude of $\Delta/8192$. Hence, a gain error of just $\alpha_1 = 1/4096$ is sufficient to cause the resulting pipelined ADC error to be comparable in magnitude to the pipelined ADC's quantization error. More generally, if $\alpha_n = (\Delta/2)^{1-n}/4096 = 2^{n-13}\Delta^{1-n}$, the nth term in (5) gives rise to an error component in the pipelined ADC output with a magnitude comparable to the pipelined ADC's quantization error.

The 15-bit, 40-MS/s pipelined ADC integrated circuit (IC) presented in [18] provides a convenient circuit-level example of the issues described above. The ADC is based on the architecture shown in Fig. 1 modified to include digital background calibration techniques that cancel ADC error arising from DAC capacitor mismatches and interstage gain errors. The ADC achieves over 90 dB of spurious-free dynamic range (SFDR) and 72 dB of peak signal-to-noise-and-distortion ratio (SNDR) over the 20-MHz bandwidth. To achieve sufficiently low distortion for this level of ADC performance, high-power residue amplifiers are used in the design: the op-amps in the residue amplifiers consume approximately 80% of the 400 mW consumed by the entire IC.

Had the sample rate been higher than 40 MHz, even higher performance, and, therefore, higher power residue amplifiers would have been required to maintain the same SFDR and peak SNDR. For example, circuit simulations indicate that the pipelined ADC's SFDR and peak SNDR drop to 65 and 56 dB, respectively, if the sample rate is increased to 100 MHz without improving the performance of the residue amplifiers.[1] Simulation of the residue amplifier stage indicates that this reduction in performance comes from both linear gain error associated with incomplete settling and from third-order distortion; the use of differential circuitry causes the even-order terms to be negligible in this example relative to the target specifications of 90-dB SFDR and 72-dB peak SNDR, and, although higher order distortion terms are present, they too are negligible in this example. Later in the paper, this example is revisited and an implementation of the HDC technique is described that digitally measures and cancels the error introduced by the residue amplifiers to restore the SFDR and peak SNDR to their target values of 90- and 72-dB, respectively.

[1]The digital logic in the IC limited the clock rate to 50 MHz, so this observation had to be made via transistor-level simulation. However, circuit simulation results up to 50 MHz are consistent with measured results.

III. Signal Processing Details of HDC Technique

A. An mth-Order Distortion Correction Example

To demonstrate the basic idea underlying the HDC technique, a simplified case is considered first: the residue amplifier in the first stage introduces only mth-order distortion, i.e., $f(v_1) = \alpha_m v_1^m$ for some integer, m, and all other components in the pipelined ADC are ideal.

The HDC technique for this example is shown in Fig. 3. A set of m uncorrelated, two-level, pseudorandom, digital *calibration sequences*, $t_1[n], t_2[n], \ldots, t_m[n]$, each of which takes on values of $\pm A$, is zero-mean, and is independent of the pipelined ADC's input signal, are added to the output of the flash ADC. They are converted to analog form along with the output sequence from the flash ADC, so the input to residue amplifier at the nth sample time is

$$v_1(nT_s) = -e_{\mathrm{ADC1}}[n] - \sum_{k=1}^{m} t_k[n]. \tag{7}$$

The amplitude A of the calibration sequences is chosen such that the sum of the calibration sequences has a maximum amplitude of approximately $\Delta/4$. Since the sum of the calibration sequences is amplified along with the quantization error from the flash ADC, this implies that approximately half of the over-range margin is taken up by the calibration sequences, which leaves the other half of the over-range margin for error associated with nonideal circuit behavior.

The calibration sequences are subjected to the distortion function of the residue amplifier along with the quantization error from the first pipeline stage, and, by reasoning similar to that presented in the previous section to obtain (3)

$$r_1[n] = v_1(nT_s) + \alpha_m v_1^m(nT_s) + \frac{1}{4096} e_{\mathrm{ADC7}}[n]. \tag{8}$$

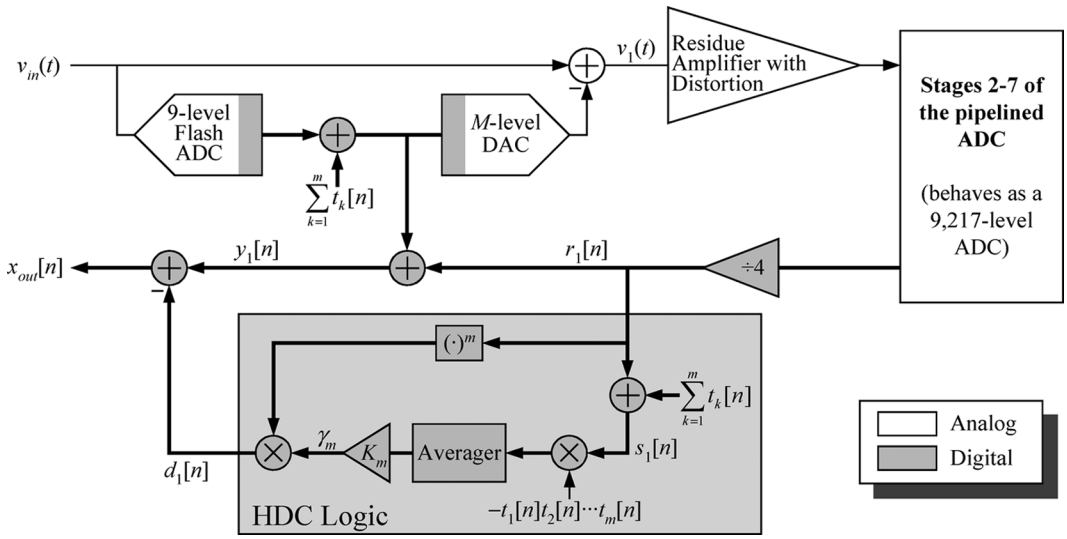

Fig. 3. Example of the HDC technique for correction of mth-order residue amplifier distortion.

It follows that the pipelined ADC output prior to correction by the HDC technique is

$$y_1[n] = v_{\text{in}}(nT_s) + \alpha_m v_1^m(nT_s) + \frac{1}{4096} e_{\text{ADC7}}[n]. \quad (9)$$

The purpose of the HDC logic is to estimate $\alpha_m v_1^m(nT_s)$ with which to cancel the mth-order distortion in $y_1[n]$, i.e., the second term in (9). It does this by correlating $r_1[n]$ against the product of the calibration sequences, $t_1[n]t_2[n]\cdots t_m[n]$. The correlation involves multiplying the digital sequence

$$\begin{aligned} s_1[n] &= r_1[n] + \sum_{k=1}^{m} t_k[n] \\ &= -e_{\text{ADC1}}[n] + \alpha_m v_1^m(nT_s) + \frac{1}{4096} e_{\text{ADC7}}[n] \end{aligned} \quad (10)$$

by $t_1[n]t_2[n]\cdots t_m[n]$, a two-level sequence that takes on values of $\pm A^m$, and averaging the result. Since the calibration sequences are zero-mean, uncorrelated with each other, and independent of the pipelined ADC's input signal, it follows that $t_1[n]t_2[n]\cdots t_m[n]$ is uncorrelated with all of the terms in (10) except the term $(m!)t_1[n]t_2[n]\cdots t_m[n]\alpha_m$ that occurs in the expansion of $v_1(nT_s)$, as given by (7), raised to the mth power. Consequently, the average of $s_1[n]$ times $t_1[n]t_2[n]\cdots t_m[n]$ over n is $(m!)A^{2m}\alpha_m$. The HDC logic multiplies the output of the averager by $K_m = A^{-2m}/(m!)$ to obtain γ_m which is an estimate of α_m. It then multiplies γ_m by $r_1^m[n]$ to obtain the estimate of $\alpha_m v_1^m(nT_s)$.

To the extent that the calibration sequences have the above-mentioned statistical properties, γ_m converges exactly to α_m as the number of samples averaged by the HDC logic increases; the more samples in the average, the better the estimate of α_m. This convergence occurs regardless of the pipelined ADC's input signal, so the HDC technique performs background calibration, i.e., it functions during normal operation of the pipelined ADC. After an initial *convergence time* during which the averager obtains a sufficiently accurate estimate of α_m that the pipelined ADC's accuracy is limited by nonideal circuit behavior other than mth-order residue amplifier distortion, the pipelined ADC operates at its full accuracy, and the HDC technique continues to track slow variations in α_m that may occur because of temperature changes or as the device ages.

Although the estimate of α_m has an accuracy that depends only upon the number of samples averaged by the HDC logic, the accuracy of the estimate of $\alpha_m v_1^m(nT_s)$ is limited by the presence of unwanted higher order terms that occur in $r_1^m[n]$. For example, it follows from (8) that if $m = 3$ and the small last term of (8) is neglected, then

$$\begin{aligned} \alpha_3 r_1^3[n] \cong\ & \alpha_3 v_1^3(nT_s) \\ & + \underbrace{3\alpha_3^2 v_1^5(nT_s) + 3\alpha_3^3 v_1^7(nT_s) + \alpha_3^4 v_1^9(nT_s)}_{\text{unwanted terms}}. \end{aligned} \quad (11)$$

Fortunately, as demonstrated in the next section, the unwanted terms in (11) tend to be small in practice in which case they can be neglected.

For the special case of $m = 1$, the HDC technique as shown in Fig. 3 reduces to the gain error correction (GEC) technique presented in [18] and [22]. Hence, the HDC technique can be viewed as an extension of the GEC technique.

B. HDC Technique for Correction of Multiple Orders of Distortion

By a minor extension of the analysis presented above, it is easy to verify that γ_m converges to α_m even if the residue amplifier's distortion function contains lower order distortion terms. In other words, even if any of the α_i for $i < m$ are nonnegligible in (5), the HDC logic shown in Fig. 3 accurately estimates α_m.

However, a complication arises if any of the α_i for $i > m$ are nonnegligible. For example, suppose that the HDC technique as shown in Fig. 3 is implemented with $m = 3$, but instead of the residue amplifier introducing only third-order distortion, it introduces first-order, third-order, and fifth-order distortion.

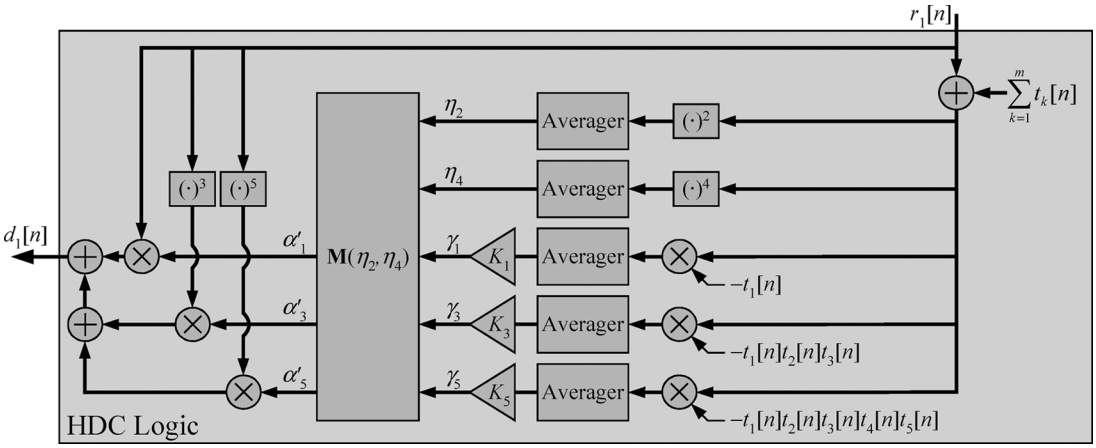

Fig. 4. An example of the HDC logic for correction of fist-order, third-order, and fifth-order residue amplifier distortion.

That is, suppose $f(v_1) = \alpha_1 v_1 + \alpha_3 v_1^3 + \alpha_5 v_1^5$. In this case, (10) becomes

$$s_1[n] = -e_{\mathrm{ADC1}}[n] + \alpha_1 v_1(nT_s) + \alpha_3 v_1^3(nT_s) + \alpha_5 v_1^5(nT_s) + \frac{1}{4096} e_{\mathrm{ADC7}}[n] \quad (12)$$

with $v_1(nT_s)$ still given by (7). Expanding the fifth-order term in (12) results in several cross-terms that are correlated with the product of the calibration sequences, $t_1[n]t_2[n]t_3[n]$. These terms cause γ_3 to converge to a value that differs from α_3. Specifically, γ_3 now converges to

$$\alpha_3 + \left[30A^2 + 10\left\langle e_{\mathrm{ADC1}}^2[n]\right\rangle\right]\alpha_5 \quad (13)$$

as the number of averaged samples increases, where $\langle e_{\mathrm{ADC1}}^2[n]\rangle$ denotes the average of $e_{\mathrm{ADC1}}^2[n]$. Therefore, the presence of nonnegligible fifth-order residue amplifier distortion prevents the version of the HDC technique shown in Fig. 3 from functioning properly because of the unwanted α_5 term in (13).

As another example, consider the same distortion function, but suppose $m = 1$. In this case the HDC logic correlates a single calibration sequence, $t_1[n]$, against $r_1[n]$ to obtain γ_1. It follows from the presentation above that in the absence of third-order and fifth-order distortion, γ_1 would converge to α_1. However, the third-order and fifth-order terms in (12) contain several cross-terms that are correlated with $t_1[n]$. Consequently, γ_1 converges to

$$\alpha_1 + \left[13A^2 + 3\left\langle e_{\mathrm{ADC1}}^2[n]\right\rangle\right]\alpha_3 + \left[241A^4 + 130A^2\left\langle e_{\mathrm{ADC1}}^2[n]\right\rangle + 5\left\langle e_{\mathrm{ADC1}}^4[n]\right\rangle\right]\alpha_5. \quad (14)$$

From these examples, it is evident that the HDC technique must be modified for cases in which the residue amplifier's distortion function has more than one nonnegligible term. The idea is to use N two-level calibration sequences as described above, correlate $r_1[n]$ against $t_1[n]t_2[n]\cdots t_k[n]$ to obtain γ_k for each $k = 1, 2, \ldots, N$, at which α_k is nonnegligible, and estimate the unwanted terms in each γ_k value to obtain an estimate of the corresponding α_k.

For example, suppose again that $f(v_1) = \alpha_1 v_1 + \alpha_3 v_1^3 + \alpha_5 v_1^5$. In this case, 5 calibration sequences are used, each of which takes on values of $\pm A$ where $A = \Delta/20$. The corresponding HDC logic is shown in Fig. 4, where $K_i = A^{-2i}/(i!)$. It calculates γ_1, γ_3, and γ_5 as described above, as well as the averages of $r_1^2[n]$ and $r_1^4[n]$ which are denoted as η_2 and η_4, respectively. By the arguments presented above, γ_1 converges to the quantity given by (14), γ_3 converges to the quantity given by (13), γ_5 converges to α_5, and η_2 and η_4 converge to $\langle e_{\mathrm{ADC1}}^2[n]\rangle$ and $\langle e_{\mathrm{ADC1}}^4[n]\rangle$, respectively. Therefore, the vector $\boldsymbol{\alpha}' = \mathbf{M}(\eta_2, \eta_4)\,\boldsymbol{\gamma}$ converges to $\boldsymbol{\alpha}$ where

$$\boldsymbol{\alpha} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}, \quad \boldsymbol{\gamma} = \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix}$$

and $\mathbf{M}(\eta_2, \eta_4)$ is shown in the equation at the bottom of the page.

The HDC logic uses the resulting estimated values of α_1, α_3, and α_5 to cancel the corresponding distortion terms in the pipelined ADC's output sequence as shown in Fig. 4.

C. Convergence Time

It follows from the presentation above that the γ_k values calculated by the HDC logic can be written as

$$\gamma_k = \frac{1}{k!A^kP}\sum_{i=0}^{P-1} s_1[i]c[i] \quad (15)$$

$$\begin{bmatrix} 1 & -13A^2 - 3\eta_2 & -241A^4 + 390A^6 + 90A^4\eta_2 + 30\eta_2^2 - 5\eta_4 \\ 0 & 1 & -30A^2 - 10\eta_2 \\ 0 & 0 & 1 \end{bmatrix}$$

where

$$c[n] = \begin{cases} 1, & \text{if } t_1[n]t_2[n]\cdots t_k[n] > 0 \\ -1, & \text{otherwise} \end{cases}$$

and P is the number of samples averaged by the averager blocks. The sign of the product of the calibration sequences, $c[n]$, is a random sequence, so for any finite value of P, γ_k is a random variable.

If the averagers in the HDC logic were ideal, they would evaluate (15) in the limit as $P \to \infty$ in which case γ_k would converge to its ideal value, $\gamma_k|_{\text{ideal}}$. However, P is finite in any practical averager, so the convergence process is incomplete and this introduces a random estimation error component. The mean squared value of the estimation error, i.e., $\mathrm{E}\{(\gamma_k - \gamma_k|_{\text{ideal}})^2\}$, can be used to quantify the estimation error. By its definition, $c[n]$ is a white random sequence with zero mean and unity variance. It is independent of the pipelined ADC's input sequence, any term that does not contain one or more of the sequences $t_1[n], t_2[n], \ldots, t_k[n]$ as factors, and any term that contains a calibration sequence other than $t_1[n], t_2[n], \ldots, t_k[n]$ as a factor. With A set to $\Delta/(4m)$ (to provide a specific example), it follows from these properties and (15) that

$$\mathrm{E}\left\{(\gamma_k - \gamma_k|_{\text{ideal}})^2\right\} = \frac{1}{P}\left(\frac{4m}{\Delta}\right)^{2k}\left(\frac{1}{k!}\right)^2\left(\frac{1}{P}\sum_{i=0}^{P-1} u_i^2[i]\right) \tag{16}$$

where m is the number of calibration sequences, and $u_1[n]$ is equal to $s_1[n]$ minus the terms that are correlated with $c[n]$. Equation (16) specifies the relationship between the number of samples averaged and the convergence accuracy of the HDC technique. By the design of the pipelined ADC, $|u_1[n]| < \Delta$, so (16), viewed as a function of P, has the form of a bounded sequence divided by P. Hence, as expected this implies that the estimation error goes to zero as $P \to \infty$.

The required convergence time is the minimum value of P for which the HDC logic is able to measure all the α_k values with sufficient accuracy that the error arising from residue amplifier distortion is canceled to the point that the target specifications of the pipelined ADC are met. Equation (16) gives insight into which terms affect the required convergence time. However, a closed-form expression for the required convergence time is not yet known. Hence, as demonstrated in the next section, computer simulations are used to determine the required convergence time on a case-by-case basis.

One insight offered by (16) is that the mean squared estimation error for a given value of P gets worse as k is increased. The number of calibration sequences, m, must at least equal the order of the highest order distortion term to be measured by the HDC logic, so m is at least as large as k in (16), and the mean squared estimation error is proportional to m^{2k}. Thus, the highest order distortion term to be measured generally determines the required convergence time. For example, in the HDC technique implementation presented in the next section, the third-order distortion term is the highest term measured by the HDC logic. This term causes the required convergence time to be on the order of four billion samples (e.g., 40 s worth of samples at a sample rate of 100 MHz).

D. Overview of Practical Issues

To simplify the presentation, the HDC technique has been described up to this point under the unrealistic assumption that the only nonideal analog component in the pipelined ADC is the residue amplifier in the first pipeline stage. However, as described in the remainder of the paper, the HDC technique is able to function effectively in the presence of realistic circuit nonidealities.

It follows from the analysis presented above that the convergence process works in the presence of any signal that is statistically independent of the calibration sequences. Therefore, circuit noise does not bias the HDC convergence process. This leaves distortion (from components other than the residue amplifier) as the only potential nonideal circuit behavior that can significantly affect the convergence of the HDC technique. For example, if the DAC in a pipeline stage to which the HDC technique is applied introduces nonnegligible, nonlinear distortion, the HDC technique will not properly correct for the residue amplifier distortion. Fortunately, with dynamic element matching (DEM) to scramble component mismatches, the DACs in a pipelined ADC can be implemented with extremely high linearity [18], [23]. Moreover, segmentation techniques can be used to create DEM DACs that handle the extra levels required to accommodate the calibration sequences with very little extra hardware complexity or latency [18], [24], [25].

In the examples presented so far, the HDC technique has been applied only to the first pipeline stage, but in general it can be applied simultaneously to as many of the pipeline stages as necessary. As can be deduced from (3) and Fig. 1, for each $k = 1, \ldots, 6$, the combination of pipeline stages k through 7 and the associated digital logic is a pipelined ADC in its own right with a resolution of approximately $2(6-k)+3$ bits. Therefore, by the reasoning presented above, the HDC technique can be applied simultaneously to any of the first six pipeline stages provided calibration sequences are used in each stage that are independent of those used in the other stages. It follows from the architecture of Fig. 1 that any distortion introduced by the kth pipeline stage is attenuated by a factor 4^{k-1} referred to the output, so the distortion introduced by all but the first few stages tends to be negligible. Consequently, in practice, it is only necessary to apply the HDC technique to the first few stages.

IV. HDC Implementation Example

An example is presented in this section in which the HDC technique is applied to the first three stages of the pipelined ADC shown in Fig. 1. The result is shown in Figs. 5–7. Fig. 5 shows a high-level view of the pipelined ADC, Fig. 6 shows the pipelined ADC with expanded views of the first pipeline stage and the associated HDC logic, and Fig. 7 shows the high-level structure of the DEM DAC used in the first three pipeline stages. The details are described below and computer simulation results are presented to demonstrate the performance of the system.

The residue amplifier distortion for this example is modeled after the behavior observed via transistor-level circuit simulations in the pipelined ADC of [18] for a sample rate of 100 MHz. Specifically, for each residue amplifier, the nonnegligible distortion terms in (5) are $\alpha_1 = -0.0125$, $\alpha_3 = -2^{-6}\Delta^{-2}$, $\alpha_5 = -2^{-9}\Delta^{-4}$, and $\alpha_7 = -2^{-11}\Delta^{-6}$, where $\Delta = 250$ mV is the

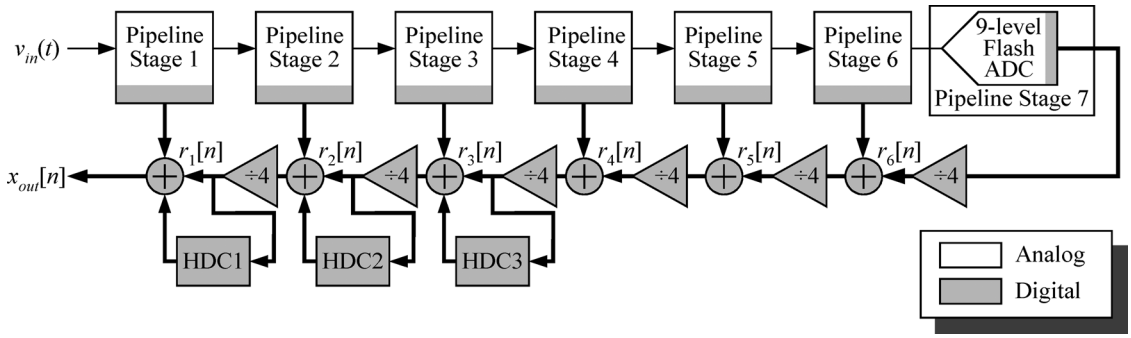

Fig. 5. High-level view of an example pipelined ADC incorporating the HDC technique.

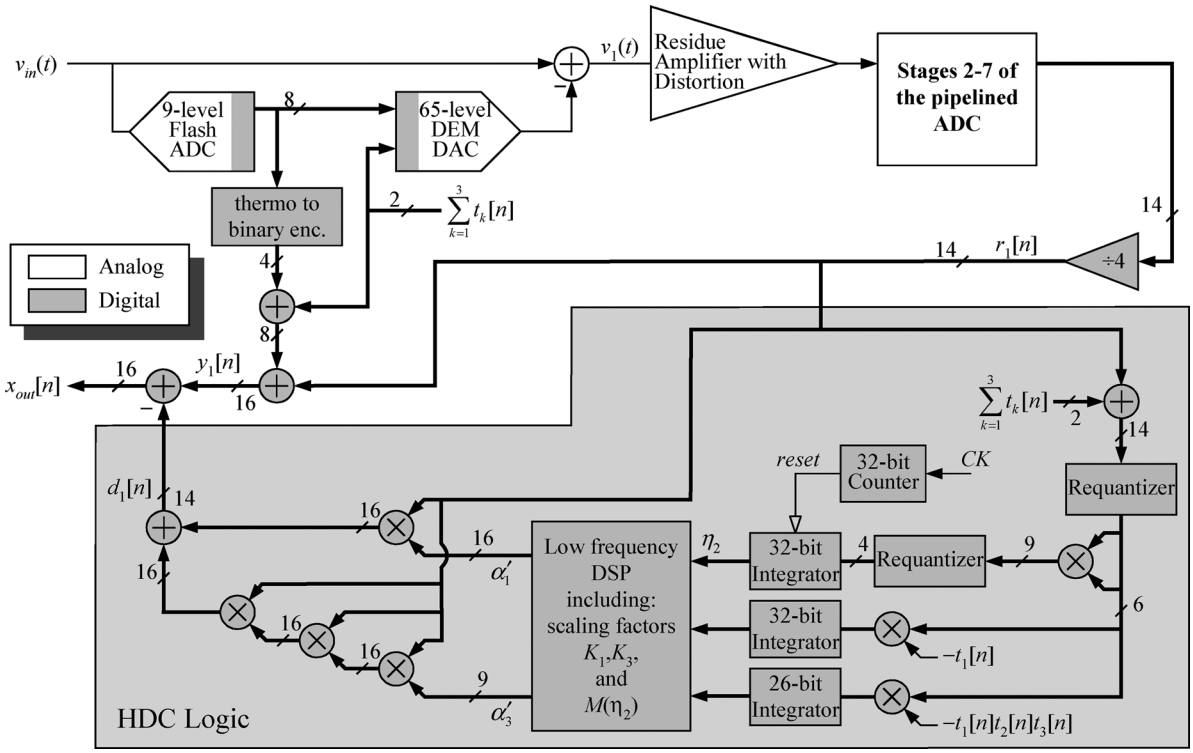

Fig. 6. Example pipelined ADC incorporating the HDC technique with expanded views of the first stage and associated HDC logic.

step size of the flash ADC. It can be deduced for this case from the results presented in Section II that only the first-order and third-order residue amplifier distortion terms in the first three pipeline stages need be cancelled to achieve 15-bit pipelined ADC accuracy. Therefore, the HDC technique is applied in this example to measure and cancel just these distortion terms.

The details of the first pipeline stage and associated HDC logic are shown in Fig. 6. Three pseudorandom $\pm\Delta/16$ calibration sequences are added prior to the DAC, so the sum of the calibration sequences is a four-level sequence that can range from $-3\Delta/16$ to $3\Delta/16$ in steps of $\Delta/8$. The use of three $\pm\Delta/16$ calibration sequences has two analog circuit implications. The first implication is that the DAC must have a minimum step size of $\Delta/8$ (instead of Δ as in the fourth through seventh pipeline stages) and enough levels to accommodate the calibration sequences. To avoid exceeding the input range of the DAC, the sum of the calibration sequences are forced to $\Delta/16$ and the HDC estimators for the pipeline stage are disabled when the output of the flash ADC is either at its maximum or minimum value. Therefore, the sum of the calibration sequences and the flash ADC output can take on values of

$$k\Delta + i\Delta/8 + \Delta/16 \tag{17}$$

where

$$k = -4, -3, \ldots, 4 \text{ and } i = \begin{cases} -2, -1, 0, 1, & \text{if } |k| \leq 3 \\ 0, & \text{if } |k| = 4 \end{cases}$$

so the DAC must be able to generate these output levels. The second implication is that the calibration sequences occupy almost half of what would otherwise have been the over-range margin. Specifically, it follows from the discussion in Section II that the over-range margin for each of the first three stages is $\pm 1.75\Delta$. While this tightens the design constraints on the flash ADC, it is not difficult to handle in practice [18].

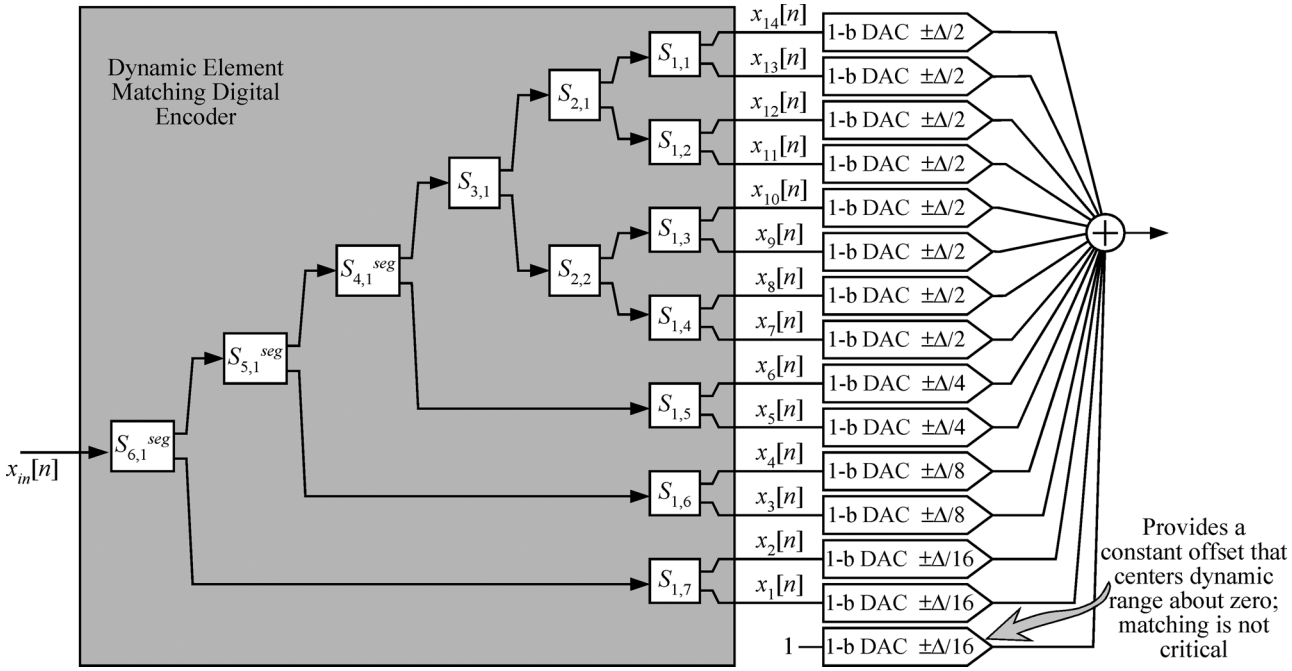

Fig. 7. Functional view of the DEM DAC.

As described above, the DAC in each of the first three stages must be capable of generating the output levels specified by (17). This is accomplished by the DAC architecture shown in Fig. 7. It consists of a digital DEM encoder block and 15 1-bit DACs. Each one-bit DAC outputs a nominal value of $-q\Delta$ or $q\Delta$ depending upon whether its input bit is 0 or 1, respectively, where q is a *weighting factor* that is fixed for a given 1-bit DAC. There are three 1-bit DACs with $q = 1/16$, two with $q = 1/8$, two with $q = 1/4$, and eight with $q = 1/2$. With this 1-bit DAC weighting arrangement, for most of the possible input values, $x_{\rm in}[n]$, there are multiple distinct bit vectors, $x_1[n], x_2[n], \ldots, x_{14}[n]$, that give rise to the desired nominal output value. At each sample clock, the DEM encoder pseudorandomly selects one of these multiple, nominally equivalent vectors.

If all the 1-bit DAC step sizes were ideal, the pseudorandom selection algorithm in the DEM encoder would have no effect. However, inadvertent component mismatches arise during circuit fabrication which causes the 1-bit step sizes to deviate from their ideal values. If only one of the possible values of the 1-bit DAC input vector, $x_1[n], x_2[n], \ldots, x_{14}[n]$, were used for each value of $x_{\rm in}[n]$, the step-size errors would cause the overall DAC to introduce harmonic distortion. By pseudorandomly choosing among the different possible 1-bit DAC input vectors for each input sample, the DEM encoder causes the overall DAC to introduce white noise that is uncorrelated with the other sequences in the pipelined ADC instead of harmonic distortion, and the white noise can be removed in the digital domain by a background calibration technique [18], [23].

From a signal processing point of view the DEM encoder can be viewed as a tree of digital logic blocks called *switching blocks* as shown in Fig. 7. Each switching block is labeled $S_{k,r}$ or $S_{k,r}^{\rm seg}$ in the figure, where k and r denote the position of the switching block in the tree. The three switching blocks labeled $S_{k,r}^{\rm seg}$ in the figure are called *segmented* switching blocks because in each case their two outputs affect the input bits to 1-bit DACs with different weighting factors. The ten switching blocks labeled $S_{k,r}$ are called *nonsegmented* switching blocks because in each case their two outputs only affect the input bits to 1-bit DACs with equal weighting factors.

Each switching block operates on a digital input sequence and generates two digital output sequences. The output sequences generated by each segmented switching block, $S_{k,r}^{\rm seg}$, are given by

$$x_{k-1,1}[n] = \frac{x_{k,r}[n] + s_{k,r}[n]}{2}, \text{ and } x_{1,k+1}[n] = -s_{k,r}[n] \tag{18}$$

where $x_{k,r}[n]$ is the input to the switching block and $s_{k,r}[n]$ is a pseudorandom sequence, called a *switching sequence*. The switching sequence is generated as part of the switching block logic as

$$s_{k,r}[n] = \begin{cases} 0, & \text{if } x_{k,r}[n] \text{ is even} \\ \pm 1, & \text{otherwise(chosen pseudorandomly)} \end{cases} \tag{19}$$

The output sequences generated by each nonsegmented switching block, $S_{k,r}$, are given by

$$\begin{aligned} x_{k-1,2r}[n] &= \frac{x_{k,r}[n] + s_{k,r}[n]}{2} \\ x_{k-1,2r-1}[n] &= \frac{x_{k,r}[n] - s_{k,r}[n]}{2} \end{aligned} \tag{20}$$

where, as before, $x_{k,r}[n]$ is the input to the switching block and $s_{k,r}[n]$ is a switching sequence given by (19). It can be verified from the results presented in [24], [26], and [27] that the DEM encoder ensures that output level errors in the 1-bit DACs from

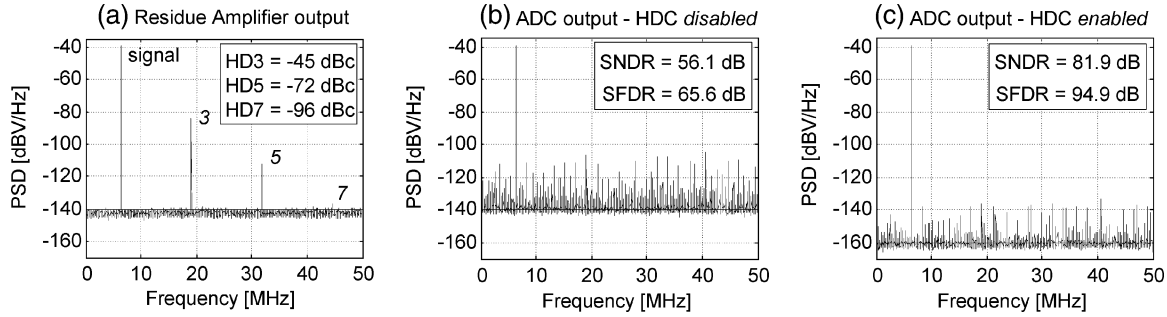

Fig. 8. Simulation results for HDC applied to first, second, and third stages.

component mismatches do not cause the overall DAC to introduce harmonic distortion, which is a requirement of the HDC technique.

It follows from (18)–(20) that the data paths through the switching blocks are not clocked, so the DEM encoder could be implemented directly as combinational logic. However, in high-speed pipelined ADCs, latency from the output of the flash ADC through the DAC in each pipeline stage must be minimized because the larger the latency the less time is available for the residue amplifier following the DAC to settle. In [18], this issue was addressed by implementing the functionality of both the calibration sequence adder and the DEM encoder in parallel as a single layer of digital transmission gates along with some digital logic gates through which latency is not critical. This reduced the latency from the output of the flash ADC through the DEM encoder to that of a single transmission gate. Although the DEM encoder shown in Fig. 7 is more complicated than that presented in [18], the same approach has been taken in the computer simulated implementation described below. Since the calibration sequences are known in advance of the flash ADC output data, only the combinational logic component through which latency is not critical is increased in this example relative to the DEM encoder presented in [18].

The practical version of the HDC logic shown in Fig. 6 is a direct implementation of ideal version shown in Fig. 4, except without fifth-order distortion correction. The primary differences between the practical and ideal versions are that requantization is used to reduce the bit widths of various data buses to reduce digital complexity, and the three averagers are implemented with $P = 2^{32}$ in the practical version. Dithered requantizers are used to perform the requantization as described in [23] to avoid introducing harmonic distortion. Requantization is not necessary, but by reducing data bus widths it greatly reduces the area and power consumption of the HDC logic, yet the quantization noise it introduces adds only slightly to the HDC convergence time. The random dither sequences and calibration sequences in this example were generated by a single linear feedback shift register of the form described in [28].

At a sample rate of 100 MHz with $P = 2^{32}$, each HDC block requires approximately 43 seconds to converge. However, the accuracy of each HDC block depends on the accuracies of the HDC blocks in the subsequent stages. Thus, the total convergence time for this example implementation is approximately 2 min.

V. Simulation Results and HDC Limitations

The example pipelined ADC with HDC as described above was simulated with various nonideal circuit effects. The simulated residue amplifier distortion in each stage includes the first through seventh-order distortion terms described above. The 1-bit DAC mismatches were chosen as independent Gaussian random variables; the standard deviations of the 1-bit DACs with step sizes of Δ, $\Delta/2$, $\Delta/4$ and $\Delta/8$ are 0.30%, 0.42%, 0.60%, 0.85%, of $\Delta = 250\ \mathrm{mV}$, respectively. The flash ADC threshold errors and residue amplifier offset voltages were chosen as independent Gaussian random variables with standard deviations of 25 and 5 mV, respectively. A 10-$\mathrm{nV}^2_{\mathrm{rms}}$ white noise signal was added at the input of each residue amplifier to model thermal noise.

Fig. 8(a) shows the power-spectral density (PSD) plot[2] of the output of the residue amplifier simulated alone with a 275 mV, 6.4-MHz sinusoidal input signal. The amplitude of the input signal is nearly the maximum input that does not overload the next stage of the pipeline. Hence, the output of the residue amplifier consists of the 6.4-MHz fundamental tone plus the residue amplifier distortion terms and thermal noise. The plot demonstrates the nonlinear behavior of the residue amplifier.

Fig. 8(b) and (c) shows the PSD plots of the pipelined ADC with a -1 dB relative to full-scale 6.4-MHz sinusoidal input signal. Fig. 8(b) shows the case with the HDC technique *disabled*, and Fig. 8(c) shows the case with the HDC technique *enabled*. Comparison of Fig. 8(b) and (c) indicates that the HDC technique improved the simulated SNDR and SFDR by 26 and 30 dB, respectively. Numerous other simulations performed by the authors with different input signals, and different random mismatches, ADC thresholds, and DAC mismatches, exhibit similar results.

Before computing the PSD estimates for the simulation results shown in Fig . 8(b) and (c), the components of the final output signal corresponding to DAC mismatches and thermal noise were removed so as not to obscure the effect of the HDC technique. Removal of the components corresponding to DAC mismatches can be achieved in a practical implementation via the DNC technique presented in [18] and [23]. However, the DNC technique is not necessary for the HDC technique to function provided DEM DACs are used to ensure that error introduced by DAC mismatches does not contain significant harmonic distortion.

One potential limitation of the HDC technique is not demonstrated by the implementation example described above. The

[2]The PSDs were estimated using 16 Hanning windowed periodograms of length 16384.

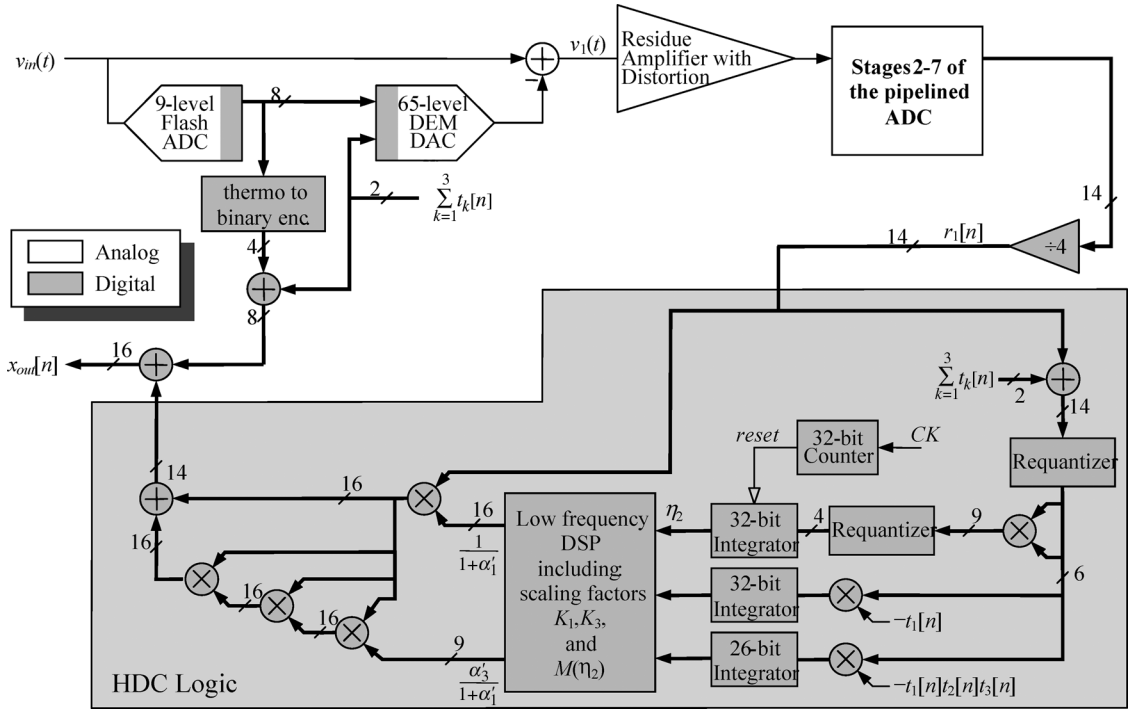

Fig. 9. Example pipelined ADC incorporating the HDC technique with improved correction scheme for large distortion coefficients.

version of the correction scheme presented in Figs. 4 and 6 is not accurate if the error is too big, i.e., if the α_n coefficients in (5) are too large—this could happen, for instance, if an open-loop residue amplifier configuration as in [19] is used instead of a classical closed-loop configuration. For example, if the distortion function given by (5) can be written as

$$f\left(v_1(nT_s)\right) = \alpha_1 v_1(nT_s) + \alpha_3 v_1^3(nT_s \tag{21}$$

and the digitized residue $r_1[n]$ in Fig. 6 is given by

$$r_1[n] \cong (1+\alpha_1)v_1(nT_s) + \alpha_3 v_1^3(nT_s) \tag{22}$$

the correction signal $d_1[n]$ is

$$\begin{aligned} d_1[n] \cong & \left(\alpha_1' + \alpha_1'^2\right) v_1(nT_s) \\ & + \left(\alpha_3' + 4\alpha_1'\alpha_3' + 3\alpha_1'^2\alpha_3' + \alpha_1'^3\alpha_3'\right) v_1^3(nT_s) \\ & + \left(3\alpha_3'^2 + 6\alpha_1'\alpha_3'^2 + 3\alpha_1'^2\alpha_3'^2\right) v_1^5(nT_s) + \ldots \end{aligned} \tag{23}$$

Subtracting (23) from the uncorrected output given by (6), using (21), and assuming that $\alpha_n \approx \alpha_n'$, the pipeline output is

$$\begin{aligned} x_{\text{out}}[n] \cong & \, x_{\text{out}}[n]|_{\text{ideal}} - \alpha_1^2 v_1(nT_s) \\ & - \left(4\alpha_1\alpha_3 + 3\alpha_1^2\alpha_3 + \alpha_1^2\alpha_3\right) v_1^3(nT_s) \\ & - \left(3\alpha_3^2 + 6\alpha_1\alpha_3^2 + 3\alpha_1^2\alpha_3^2\right) v_1^5(nT_s) \\ & - \left(3\alpha_3^3 + 3\alpha_1\alpha_3^3\right) v_1^7(nT_s) - \alpha_3^4 v_1^9(nT_s). \end{aligned} \tag{24}$$

Comparing to (24) to (6), it is clear that HDC removes most of the distortion provided the α_n coefficients are sufficiently small. However, in some applications this may not be the case, in which case the remaining unwanted terms in (24) may not be negligible for the given application. In such cases, the modified correction technique shown in Fig. 9 can be used. A similar analysis to that presented above indicates that the pipelined ADC output is now

$$\begin{aligned} x_{\text{out}}[n] \cong & \, x_{\text{out}}[n]|_{\text{ideal}} - \frac{3\alpha_3^2}{(1+\alpha_1)^2} v_1^5(nT_s) \\ & - \frac{3\alpha_3^3}{(1+\alpha_1)^3} v_1^7(nT_s) - \frac{\alpha_3^4}{(1+\alpha_1)^4} v_1^9(nT_s). \end{aligned} \tag{25}$$

Equation (25) shows that linear and third-order distortion has been removed, while the remaining unwanted terms are smaller than or comparable to the respective terms in (24). The price paid for the accuracy improvement is increased complexity. Although both schemes require the same number of multipliers, the extension of the latter scheme to correct for higher order harmonics would result in a more complex hardware.

Another limitation of the HDC technique has not been highlighted by the example presented in Section IV. Had it been necessary to apply HDC to correct fifth-order residue amplifier distortion, a problem would have arisen for the chosen pipelined ADC architecture and target specifications. Specifically, the fifth-order distortion term for this case is so small that high-order distortion from the coarse quantization performed by the flash ADCs in each stage becomes significant and distorts the HDC technique's estimate. Equation (12) represents the signal used to estimate the first stage's residue amplifier distortion terms under the assumption that the following stages are either ideal or perfectly corrected. A more accurate expression for $s_1[n]$ is

$$\begin{aligned} s_1[n] \cong & -e_{\text{ADC1}}[n] + \alpha_1 v_1(nT_s) + \alpha_3 v_1^3(nT_s) + \alpha_5 v_1^5(nT_s) \\ & + \sum_{k=2}^{7} \lambda_k e_{\text{ADC}k}[n] + \ldots \end{aligned} \tag{26}$$

where the γ_k is the amplitude of uncanceled flash ADC error from the kth stage. Therefore, in the absence of perfect cancellation,

every flash ADC contributes error in (26). The error is largely quantization noise which tends to be highly correlated with $v_1[n]$ and therefore with the pseudorandom sequences. The smaller the α_n coefficients to be estimated by the HDC technique, the more significantly the imperfectly cancelled flash ADC errors distort the estimated coefficient values. Furthermore, the coarse quantization performed by the flash ADCs is a *hard nonlinearity*, so it can not be represented by a small number of Taylor series terms. In conclusion, the HDC technique, as well as any other scheme (e.g., [20]) that assumes the nonlinearity to be estimated is well-modeled by a small number of Taylor series terms fails to work well when the nonlinearity to be estimated is very small. In principle, an analog dither signal can be added prior to the flash ADCs to eliminate this problem in cases where very small distortion terms must be measured by the HDC technique [29].

REFERENCES

[1] W. T. Colleran and A. A. Abidi, "A 10-b, 75-MHz two-stage pipelined bipolar A/D converter," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, pp. 1187–1199, Dec. 1993.

[2] K. Sone, Y. Nishida, and N. Nakadai, "A 10-b 100-Msample/s pipelined subranging BiCMOS ADC," *IEEE J. Solid-State Circuits*, vol. 28, no. 12, pp. 1180–1186, Dec. 1993.

[3] J. Li and U.-K. Moon, "A 1.8-V 67-mW 10-bit 100-MS/s pipelined ADC using time-shifted CDS technique," *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1468–1476, Sep. 2004.

[4] J.-B. Park, S.-M. Yo, S.-W. Kim, Y.-J. Cho, and S.-H. Lee, "A 10-b 150-MSample/s 1.8-V 123-mW CMOS A/D converter with 400-MHz input bandwidth," *IEEE J. Solid-State Circuits*, vol. 39, no. 8, pp. 1335–1337, Aug. 2004.

[5] 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 *Dig. Tech. Papers ISSCC*, Feb. 2000, pp. 38–39.

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

[7] C. R. Grace, P. J. Hurst, and S. H. Lewis, "A 12-bit 80-MSample/s pipelined ADC with bootstrapped digital calibration," *IEEE J. Solid-State Circuits*, vol. 40, no. 5, pp. 1038–1046, May 2005.

[8] M.-J. Choe, B.-S. Song, and K. Bacrania, "A 13-b 40-MSamples/s CMOS pipelined folding ADC with background offset trimming," *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1781–1790, Dec. 2000.

[9] P. C. Yu, S. Shehata, A. Joharapurkar, P. Chugh, A. Bugeja, X. Du, S. U. Kwak, Y. Papantonopoulous, and T. Kuyel, "A 14 b 40 MSample/s pipelined ADC with DFCA," in *Dig. Tech. Papers ISSCC*, Feb. 2001, pp. 136–137.

[10] K. Nair and R. Harjani, "A 96-dB SFDR 50 MS/s digitally enhanced CMOS pipeline A/D converter," in *Dig. Tech. Papers ISSCC*, Feb. 2004, pp. 456–457.

[11] A. Zanchi and F. Tsay, "A 16-bit 65-MS/s 3.3-V pipeline ADC core in SiGe BiCMOS with 78-dB SNR and 180-fs jitter," *IEEE J. Solid-State Circuits*, vol. 40, no. 6, pp. 1225–1237, Jun. 2005.

[12] H.-C. Liu, Z.-M. Lee, and J.-T. Wu, "A 15-b 40-MS/s CMOS pipelined analog-to-digital converter with digital background calibration," *IEEE J. Solid-State Circuits*, vol. 40, no. 5, pp. 1047–1056, May 2005.

[13] S.-T. Ryu, S. Ray, B.-S. Song, G.-H. Cho, and K. Bacrania, "A 14-b linear capacitor self-trimming pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 39, no. 11, pp. 2046–2051, Nov. 2004.

[14] Y. Chiu, P. Gray, and B. Nikolic, "A 1.8 V 14 b 10 MS/s pipelined ADC in 0.18-μm CMOS with 99-dB SFDR," in *Dig. Tech. Papers ISSCC*, Feb. 2004, pp. 458–459.

[15] A. M. Abo and P. R. Gray, "A 1.5-V, 10-bit, 14.3-MS/s CMOS pipeline analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 34, no. 5, pp. 599–606, May 1999.

[16] I. Mehr and L. Singer, "A 55-mW, 10-bit, 40-Msample/s Nyquist-rate CMOS ADC," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 318–325, Mar. 2000.

[17] Y. Chiu, C. W. Tsang, B. Nikolic, and P. R. Gray, "Least mean square adaptive digital background calibration of pipelined analog-to-digital converters," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 51, no. 1, pp. 38–46, Jan. 2004.

[18] E. Siragusa and I. Galton, "A digitally enhanced 1.8 V 15 b 40 MS/s CMOS pipelined ADC," *IEEE J. Solid-State Circuits*, vol. 39, no. 12, pp. 2126–2138, Dec. 2004.

[19] B. Murmann and B. Boser, "A 12 b 75 MS/s Pipelined ADC using open-loop residue amplification," *IEEE J. Solid-State Circuits*, vol. 38, no. 12, pp. 2040–2050, Dec. 2003.

[20] J. P. Keane, P. J. Hurst, and S. H. Lewis, "Background interstage gain calibration technique for pipelined ADCs," *IEEE Trans. Circuits Syst. I, Reg.Papers*, vol. 52, no. 1, pp. 32–43, Jan. 2005.

[21] S. H. Lewis and P. R. Gray, "A pipelined 5-Msample/s 9-bit analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. SC-22, pp. 954–961, Dec. 1987.

[22] E. J. Siragusa and I. Galton, "Gain error correction technique for pipelined analogue-to-digital converters," *Electron. Lett.*, vol. 36, no. 7, pp. 617–618, Mar. 30, 2000.

[23] I. Galton, "Digital cancellation of D/A converter noise in pipelined A/D converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 47, no. 3, pp. 185–196, Mar. 2000.

[24] A. Fishov, E. Siragusa, J. Welz, E. Fogleman, and I. Galton, "Segmented mismatch-shaping D/A conversion," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2002, vol. 4, pp. 679–682.

[25] S. Pamarti, L. Jansson, and I. Galton, "A wideband 2.4-GHz delta–sigma fractional-N PLL with 1-Mb/s in-loop modulation," *IEEE J. Solid-State Circuits*, vol. 39, no. 1, pp. 49–62, Jan. 2004.

[26] I. Galton, "Spectral shaping of circuit errors in digital-to-analog converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 10, pp. 808–817, Oct. 1997.

[27] J. Welz and I. Galton, "Necessary and sufficient conditions for mismatch shaping in a general class of multibit DACs," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 49, no. 12, pp. 748–759, Dec. 2002.

[28] E. Fogleman, I. Galton, W. Huff, and H. Jensen, "A 3.3-V single-poly CMOS audio ADC delta–sigma modulator with 98-dB peak SINAD and 105-dB peak SFDR," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 297–307, Mar. 2000.

[29] A. B. Sripad and D. L. Snyder, "A necessary and sufficient condition for quantization errors to be uniform and white," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-25, no. 5, pp. 442–448, Oct. 1977.

Andrea Panigada (S'05) received the Laurea degree in electrical engineering from the University of Pavia, Pavia, Italy, in 1999. Since 2005, he has been working toward the Ph.D. degree at the University of California, San Diego (UCSD).

From 2000 to 2004, he was a Design Engineer at the Studio di Microelettronica, Pavia, Italy, a Design Center founded by STMicroelectronics with the cooperation of the Electronic Department of the University of Pavia. There he conducted research in algorithms for the digital calibration of analog to digital converters and in the design of CMOS prototypes of sigma–delta and pipelined ADCs. In 2003, he spent one year as a Visiting Scholar at UCSD, where he worked with the Integrated Signal Processing Group (ISPG). He is currently with the ISPG. His research interests are in the field of mixed-signal integrated circuits and systems, including data converters.

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

Since 1996, he has been a Professor of Electrical Engineering at the University of California (UC), San Diego, where he teaches and conducts research in the field of mixed-signal integrated circuits and systems for communications. Prior to 1996, he was with UC Irvine, the NASA Jet Propulsion Laboratory, Acuson, and Mead Data Central. His research involves the invention, analysis, and integrated circuit implementation of critical communication system blocks such as data converters, frequency synthesizers, and clock recovery systems. In addition to his academic research, he regularly consults at several semiconductor companies and teaches industry-oriented short courses on the design of mixed-signal integrated circuits.

Dr. Galton has served on the Corporate Board of Directors, on several Corporate Technical Advisory Boards, as the Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: ANALOG AND DIGITAL SIGNAL PROCESSING, as a member of the IEEE Solid-State Circuits Society Administrative Committee, as a member of the IEEE Circuits and Systems Society Board of Governors, and as a member of the IEEE International Solid-State Circuits Conference Technical Program Committee.