A Four-Channel Time-Interleaved ADC With Digital Calibration of Interchannel Timing and Memory Errors

Chi Ho Law, *Student Member, IEEE*, Paul J. Hurst, *Fellow, IEEE*, and Stephen H. Lewis, *Fellow, IEEE*

***Abstract*—An 11-bit 160-MS/s four-channel time-interleaved double-sampled pipelined ADC implemented in a 0.35-μm CMOS process is described. Digital calibration is used to correct mismatch errors between channels as well as the memory errors that arise from the use of double sampling. The signal-to-noise-and-distortion ratio is improved from 45 to 62 dB after calibration with an 8.7-MHz input. The spurious-free dynamic range is increased from 47 dB to 79 dB.**

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

I. Introduction

TIME-INTERLEAVING parallel channels can increase the maximum sampling rate of analog-to-digital converters (ADCs) [1]–[11]. However, the performance is sensitive to offset mismatches, gain mismatches, and sample-time errors between the interleaved channels. Digital calibration to overcome offset, gain and timing mismatches in a two-channel time-interleaved ADC has been demonstrated [2], but the techniques in that paper for calibration of gain mismatch and timing error cannot be extended to more than two channels. Also, time-interleaved ADCs with double sampling can reduce the area and power dissipation [12], but sharing operational amplifiers (opamps) across channels can introduce memory errors from finite gain not previously overcome by calibration.

To demonstrate solutions to all of these problems, this paper presents a four-channel parallel pipelined ADC that consists of two pairs of double-sampled channels. It uses digital calibration to reduce the effects of gain and timing mismatches between interleaved channels as well as memory errors between double-sampled channels. The remainder of this paper is divided into five sections. Section II gives a brief review of time-interleaved ADCs with double sampling. Section III describes the channel mismatch and memory error problems as well as their digital calibration. Section IV shows the implementation of the prototype. Section V presents the measured results, and, finally, Section VI gives the conclusion.

Manuscript received February 26, 2010; revised June 20, 2010; accepted July 09, 2010. Date of current version September 24, 2010. This paper was approved by Associate Editor Michael Flynn. This work was supported in part by Analog Devices Inc., Broadcom Corporation, Linear Technology Corporation, Marvell Semiconductor Inc., Northrop Grumman Corporation, the University of California through UC MICRO Grant 08-052, and the Agilent Foundation.

C. H. Law was with the University of California, Davis, CA 95616 USA. He is now with Exar Corporation, Fremont, CA 94538 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).

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

Digital Object Identifier 10.1109/JSSC.2010.2061630

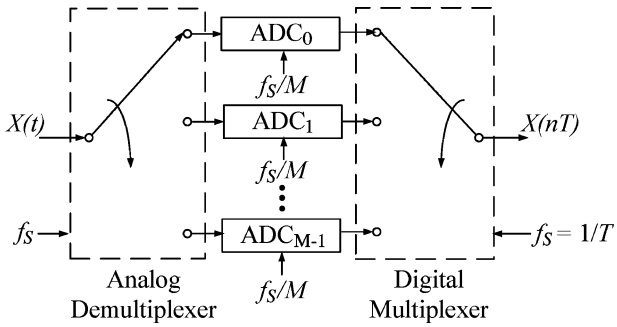

Fig. 1. Block diagram of time-interleaved ADC architecture.

II. Time-Interleaved ADC Architecture

Fig. 1 shows a simplified block diagram of a time-interleaved ADC. It consists of an analog demultiplexer at the input, M ADCs in parallel, and a digital multiplexer at the output. Each channel operates at the overall sampling rate f_s divided by M. During operation, the analog demultiplexer selects each channel in turn to process the input signal. The digital multiplexer selects the digital output of the corresponding channel. The main advantage of this structure is that it increases the overall sampling rate in a given process technology by a factor of M. Unfortunately, the performance of time-interleaved ADCs is sensitive to mismatches. Also, this architecture increases the area and power dissipation by a factor of M. One way to reduce the area and power dissipation of a time-interleaved ADC is to use double sampling.

Fig. 2 shows a block diagram of a two-channel double-sampled pipelined ADC [12]. Each channel is divided into stages, and the stages use opamps to form parasitic-insensitive sample-and-hold amplifiers. Traditionally, each opamp is used in feedback in a given stage during only one of the two nonoverlapping phases. As a result, using an opamp per stage and having the opamps do nothing half the time wastes power. Since the two channels here operate 180° out of phase with respect to each other, the opamps can be shared across channels in corresponding stages. For example, opamp a_1 is used in stage 1 of channel A during $\Phi 1$, and then it is used by stage 1 of channel B during $\Phi 2$. This structure reduces the number of required opamps by a factor of two, saving area and power dissipation

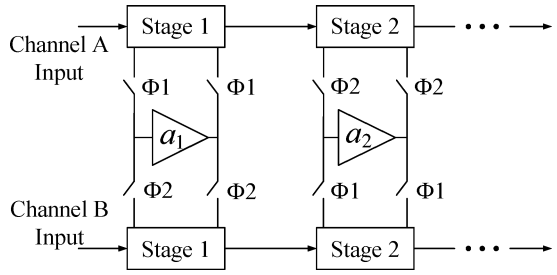

Fig. 2. Block diagram of a two-channel pipelined ADC with double sampling.

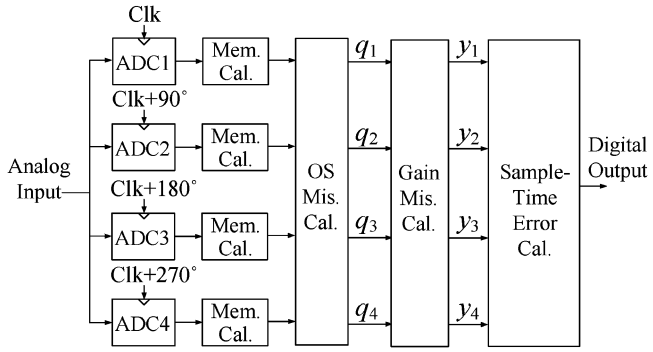

Fig. 3. Block diagram of the adaptive calibration system.

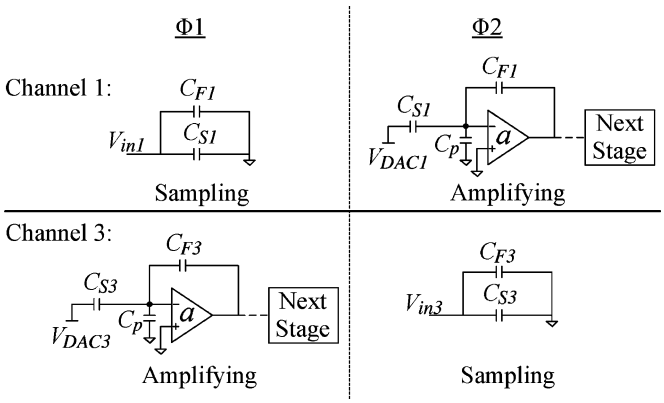

Fig. 4. Operation of the double-sampled multiplying DAC architecture.

as a result. Unfortunately, this arrangement does not allow time to reset the opamp inputs. As a result, with nonideal opamps, the signal in one channel affects the other and *vice versa* through a memory effect that is described in Section III.

Another problem in time-interleaved ADCs with or without double sampling is mismatch between the channels. Offset mismatches among the ADC channels contribute to a periodic additive pattern in the output of the ADC array. In the frequency domain, this pattern appears as tones at integer multiples of the channel sampling rate. Gain mismatches between the parallel channels cause amplitude modulation of the input samples by the sequence of channel gains. In the frequency domain, this mismatch causes copies of the input signal spectrum to appear centered around integer multiples of the channel sampling rate. Ideally, each channel should sample T seconds after the previous channel. Deviations from the ideal sampling time cause phase modulation in the sampled signal. In the frequency domain, this error produces copies of the input signal spectrum at the same frequencies as the spurious components stemming from gain mismatch. All of these mismatches raise the noise floor of the ADC, thus reducing the overall signal-to-noise ratio (SNR) [1]–[11].

III. Digital Calibration

One way to improve the performance of a time-interleaved ADC is through calibration. Analog [13] and mixed-signal calibration techniques [14], [15] that reduce timing errors have been published. These techniques adjust the sampling instants of the individual channels using analog delay cells. This approach needs small die area and power dissipation but also requires careful analog design to avoid introducing an unacceptable amount of jitter. On the other hand, digital calibration of timing errors uses digital filters to calculate ADC outputs between the sampling instants without changing the sampling clocks, potentially avoiding jitter problems and taking advantage of scaling predicted by Moore's law to reduce the area and power overhead of the digital circuits [2], [3], [6], [8]. In [2], random-chopper-based offset calibration is used to remove offset mismatches, and this technique can be extended to any number of channels. The gain and timing errors are removed by correlation-based calibration algorithms, which use the input signal itself for calibration. However, the gain and timing calibration techniques described in [2] are limited to time-interleaved ADCs with only two channels. Also, memory errors along a pipelined ADC caused by dielectric absorption/relaxation, incomplete stage reset, and the sharing of opamps have been previously analyzed [16]. This paper describes extensions of channel mismatch calibration techniques to more than two channels and extends the analysis and calibration of memory errors to include errors across channels in a time-interleaved ADC.

Fig. 3 shows a block diagram of the adaptively calibrated time-interleaved ADC system. It consists of four parallel pipelined ADCs. Each ADC operates at a quarter of the overall sampling rate, and the clock for each channel is delayed by 90° with respect to the previous channel. Notice that channels 1 and 3, as well as 2 and 4, operate 180° out of phase with respect to each other; therefore, double sampling is used within these two pairs of channels. Because of the memory-effect error stemming from double sampling, the output of each channel is followed by a memory calibration block. Then, each channel is calibrated for offset (OS) mismatch, gain mismatch, and sample-time error, respectively, to produce the overall ADC output. These calibration blocks are described in detail in Section III-A–C.

A. Interchannel Memory Error Calibration

Fig. 4 shows the operation of the double-sampled multiplying digital-to-analog converter (DAC) architecture. Single-ended circuits are shown here for simplicity. With double sampling, one opamp can be shared between two channels. For example, Fig. 4 considers one opamp that is shared between channels 1 and 3 in a four-channel case. During Φ1, channel 1 samples its input while channel 3 uses the opamp to amplify the residue from the previously sampled input. Then during Φ2,

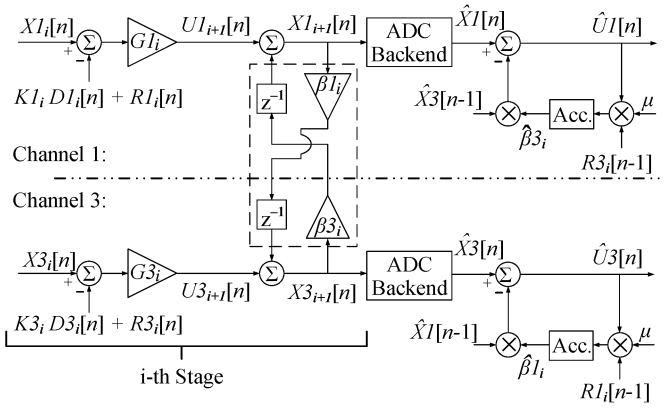

Fig. 5. Model of the memory errors in stage i and the calibration technique to overcome their effect.

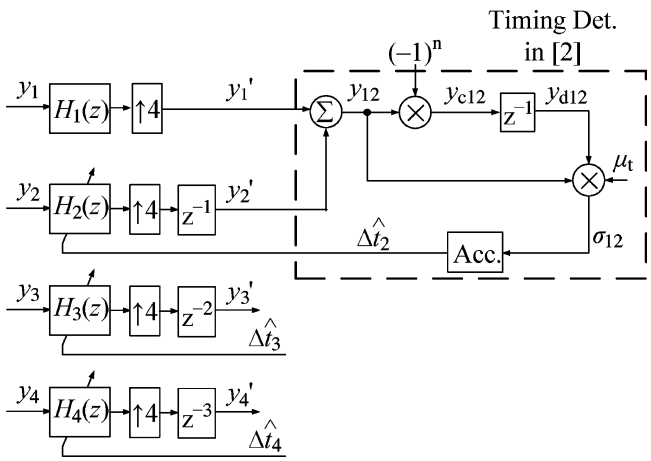

Fig. 6. Block diagram of the timing detector for channel 2 using a technique based on [2].

the operation is switched; channel 3 samples its input while channel 1 uses the same opamp to amplify the residue from the input it sampled during $\Phi 1$. The memory effect error comes from the finite gain and nonzero parasitic input capacitance of the opamp. In channel 3, finite opamp gain causes a nonzero input-dependent voltage to appear on the negative input node of the opamp. This voltage is stored on the parasitic capacitance C_p at the end of $\Phi 1$. When the opamp is then used in channel 1 during $\Phi 2$, charge conservation on this node including the charge stored on the parasitic capacitance from the previous phase determines the output. Similarly, nonzero opamp input voltage at the end of this phase introduces an error into channel 3 during the next $\Phi 1$. This coupling between channels causes a memory error that can limit performance. To overcome this limitation, the opamp inputs can be reset briefly between phases, but the reset interval would reduce the ADC conversion rate by reducing the fraction of the conversion period allocated to the sampling and amplifying operations shown in Fig. 4. Also, the reset switches would add parasitic capacitance to the opamp inputs, reducing the feedback factor around each amplifier and requiring increased power dissipation to reach a given speed. The need for reset switches can be avoided by designing the opamp to have high enough gain to avoid performance limitations stemming from this double-sampling-related memory effect [17], but this approach increases opamp complexity and power dissipation. In this project, digital background calibration is used to overcome the interchannel memory errors without requiring analog hardware that might limit performance. In fact, the analog hardware required for interchannel memory calibration is often already included for interstage gain calibration.

Fig. 5 shows a model of the memory errors in stage i of channels 1 and 3 as well as the calibration technique to overcome their effects. Signals $X1_i$ and $X3_i$ are the analog stage inputs. $K1_i$ and $K3_i$ are the gains of the digital-to-analog subconverters (DASCs), $D1_i$ and $D3_i$ are the digital input words feeding the DASCs, and $G1_i$ and $G3_i$ are the interstage gains in stage i of the pipelines. The memory effect is modeled in the dashed box. $\beta 1_i$ and $\beta 3_i$ are the ith-stage memory effect coefficients for coupling from channels 1-to-3 and 3-to-1, respectively. With memory effect, $\beta 1_i$ and $\beta 3_i$ are nonzero, and the residue voltages are coupled between channels. Since the residue is a nonlinear function of the ADC input, this coupling introduces nonlinearity in the overall ADC output. The calibration algorithm uses correlation to detect the memory coefficients. First, unique, known, and uncorrelated random sequences $R1_i$ and $R3_i$ (each with a peak-to-peak amplitude of 1/8 of full scale) are added as calibration signals to the DASC outputs of stage i in channels 1 and 3. If the memory effect coefficients are nonzero, the random sequence from one channel appears in the output of the other channel. To detect the memory effect errors, the random sequence injected in one channel is correlated with the digital ADC output of the other channel. For example, assuming the ADC backend has zero delay and memory error causes a unit delay between channels, $R1[n]$ is injected in the analog domain in stage 1 of channel 1, and $R1[n-1]$ is used in the digital domain to detect correlation between it and the output $\hat{U}3$ of channel 3. Then the estimated memory error coefficient $\hat{\beta}1_i$ is adapted through negative feedback until the correlation between $\hat{U}3$ and $R1_i$ is zero. The estimated memory error coefficient $\hat{\beta}3_i$ is found in a similar way. After adaptation, the two accumulator outputs are equal to the memory error coefficients $\beta 1_i$ and $\beta 3_i$, and their effects are removed in the channel outputs $\hat{U}1$ and $\hat{U}3$. In practice, convergence of the memory estimates requires about 50 million samples.

Consider channel 1 in Fig. 5. Signal $\hat{U}1$ can be used to reconstruct the stage input $X1_{\rm i}$ by scaling by $1/G1_i$ and adding the coarse digital representation $D1_i$. A similar approach can be used to find the previous stage input $X1_{\rm i-1}$ if memory calibrations are also performed on the $(i-1)$th stage. This approach can be repeated until the ADC input is found.

B. Sample-Time Error Calibration

The timing calibration proposed in [2] is limited to only two channels. To demonstrate this limitation, Fig. 6 shows a simple but inadequate extension of the timing calibration in [2]. Here, filtering is done before upsampling to allow the use of simple fractional delay filters [18] to correct for timing errors. Signals y_1–y_4 are the outputs of channels 1–4 after interchannel memory, offset, and gain errors have been removed. First, y_1 goes through a fixed delay finite-impulse-response (FIR) filter

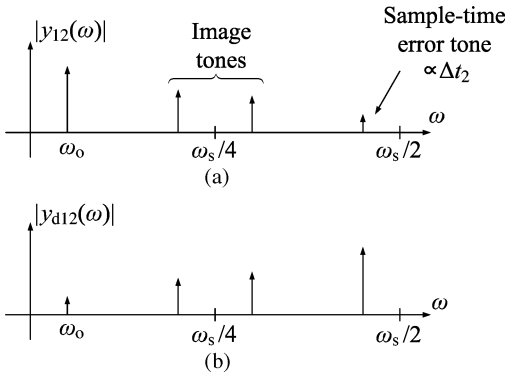

Fig. 7. Magnitude spectra with a sinusoidal input at ω_0 and $\Delta t_2 \neq 0$ for (a) y_{12} and (b) $y_{\mathrm{d}12}$.

H_1. For simplicity, initially ignore H_1 (i.e., assume $H_1(z) = 1$). Samples y_2, y_3, and y_4 go through H_2, H_3, and H_4, respectively. These filters are all-pass filters that have adaptive fractional delays [18]. For example, H_2 is controlled by $\Delta\hat{t}_2$, which is the estimate of the timing error in channel 2 found by the timing error detector in [2], introducing a delay of $\Delta\hat{t}_2/T$. Similarly, H_3 and H_4 are controlled by $\Delta\hat{t}_3$ and $\Delta\hat{t}_4$ and introduce delays of $\Delta\hat{t}_3/T$ and $\Delta\hat{t}_4/T$, respectively, but the corresponding timing error detectors for these timing errors are not shown for simplicity. The filter outputs are then upsampled by a factor of four and time-aligned to produce $y_1' - y_4'$. Then y_1' and y_2' go into the $\Delta\hat{t}_2$ timing detector block. Inside this detector, samples from channels 1 and 2 are added to produce y_{12}. This signal is then multiplied with a chopped and delayed version of itself and scaled by μ_{t} to produce σ_{12}. Then σ_{12} is sent to an accumulator, which finds an estimate of the timing error in channel 2 and controls the delay of filter H_2.

Fig. 7 shows the magnitude spectra of y_{12} and $y_{\mathrm{d}12}$ with a sinusoidal input at $\omega_0 < \omega_{\mathrm{S}}/8$ and a nonzero sample time error in channel 2. Nonzero sample-time error causes an error tone to appear at $\omega_{\mathrm{S}}/2 - \omega_0$. If the sample time error is small, the amplitude of this tone is proportional to Δt_2. This error tone is 90° out of phase with respect to the input tone. Also, since each channel has been up-sampled by a factor of 4 and y_{12} contains only samples from channels 1 and 2, image tones appear at $\omega_{\mathrm{S}}/4 \pm \omega_0$. Non-zero sample time error alters the amplitude and phase components of these image tones. Signal $y_{\mathrm{d}12}$ is a chopped and delayed version of y_{12}. The chopping causes the input and sample-time error tones in y_{12}, as well as the image tones, to reverse locations in $y_{\mathrm{d}12}$. Delaying is done to eliminate the orthogonality between the input and the error tones. The detection is based on multiplying y_{12} and $y_{\mathrm{d}12}$, and the product of the input and error tones at ω_0 and $\omega_{\mathrm{S}}/2 - \omega_0$ in the two signals produces a nonzero dc term unless the sample-time error has been corrected. Unfortunately, the products of the image tones and the corresponding chopped image tones also produce a nonzero dc component. Since this dc component is not eliminated by adjusting the estimate of Δt_2, it interferes with sensing Δt_2.

The cause and solution to this problem are explained qualitatively next. Fig. 8(a) shows a model of a time-interleaved ADC plus post processing. Assume the input V_{in} is sinusoidal

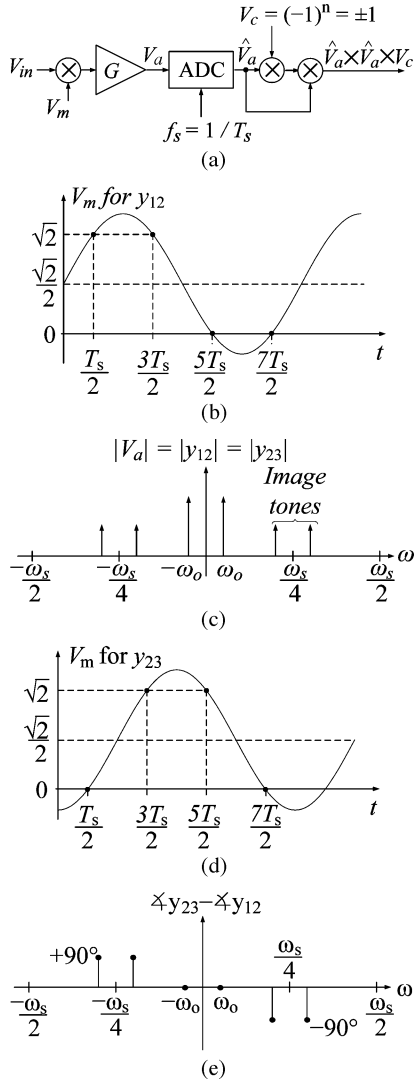

Fig. 8. (a) Model of time-interleaved ADC with sinusoidal input V_{in} at $\omega_0 < \omega_{\mathrm{S}}/8$. The ADC samples V_a at $T_s/2$, $3T_s/2$, $5T_s/2$, and $7T_s/2$. (b) Plot of V_m versus time for y_{12}. (c) Ideal magnitude spectrum of y_{12} and y_{23}. (d) Plot of V_m for y_{23}. (e) Plot of the change in the phase spectrum for y_{23} compared with y_{12}.

at $\omega_0 < \omega_{\mathrm{S}}/8$. It is multiplied by a modulating signal V_m and scaled by a constant gain G to produce the ADC input V_a. The ADC samples at $f_s = 1/T_s$, which is the full sample rate of the entire time-interleaved ADC. Assume that the ADC samples V_a at $T_s/2$, $3T_s/2$, $5T_s/2$, and $7T_s/2$. The ADC output $\widehat{V}_a$ is a quantized version of V_a. $\widehat{V}_a$ is chopped by multiplying by $V_c = (-1)^{\mathrm{n}} = \pm 1$. Then the chopper output is multiplied by the ADC output to produce the overall output of this model. V_m is selected so that the ADC takes four samples per period of V_m, one for each channel in the ADC. Therefore, V_m contains a component at $\omega_{\mathrm{S}}/4$.

At first, the goal is for $\widehat{V}_a$ to model y_{12} in Fig. 6, which includes the samples from channels 1 and 2 but excludes the samples from channels 3 and 4. So, V_m should be chosen so that $V_a = V_{\mathrm{in}}$ at $T_s/2$ and $3T_s/2$ but $V_a = 0$ at $5T_s/2$ and $7T_s/2$. Fig. 8(b) shows a plot of V_m for the y_{12} case, where $V_m = \sqrt{2}/2 + \sin(\omega_{\mathrm{S}} t/4)$. The four samples of V_m at the sample times mentioned above are marked with dots. At $T_s/2$

and $3T_s/2$, $V_m = \sqrt{2}$. At $5T_s/2$ and $7T_s/2$, $V_m = 0$. To make $V_a = V_{\rm in}$ at $T_s/2$ and $3T_s/2$, $G = 1/\sqrt{2}$ in Fig. 8(a).

Fig. 8(c) shows the ideal spectrum of $|y_{12}|$. The product of $V_{\rm in}$ at ω_0 and the component of V_m at $\omega_{\rm S}/4$ causes image tones to appear at $\omega_{\rm S}/4 \pm \omega_0$ from amplitude modulation. Since these tones appear in this ideal case (without timing error), the detector in Fig. 6 is inadequate for calibrating timing error in a four-channel time-interleaved ADC, even though it works in the two-channel case [2]. To overcome this problem, samples from channel 3 are also used.

Let y_{23} represent the sum of samples from channels 2 and 3, that is, $y_{23} = y_2' + y_3'$ in Fig. 6. To have $\widehat{V}_a$ model y_{23}, the goal in Fig. 8(a) is to set $V_a = 0$ at times that correspond to the samples in channels 1 and 4. Fig. 8(d) shows a plot of V_m for the y_{23} case, where $V_m = \sqrt{2}/2 - \cos(\omega_{\rm S} t/4)$. The key point is that the sinusoidal component of V_m here is shifted by -90° compared to the case for y_{12}. Fig. 8(c) shows the resulting spectrum of $|y_{23}|$, which is the same as for $|y_{12}|$.

From a phase standpoint, the phase at ω_0 is the same for y_{23} as for y_{12}. This result stems from the fact that V_m for both y_{12} and y_{23} contains a dc term $(\sqrt{2}/2)$ that is multiplied by the sinusoidal input at ω_0, making one part of V_a proportional to this sine wave for both y_{12} and y_{23}. However, the phase at $\omega_{\rm S}/4 \pm \omega_0$ is not the same in both cases. In the y_{12} case, the product of sine waves at $\omega_{\rm S}/4$ and ω_0 gives cosines at $\omega_{\rm S}/4 \pm \omega_0$. For y_{23}, however, the sinusoidal component at $\omega_{\rm S}/4$ is shifted by -90°, and this change causes the cosines at $\omega_{\rm S}/4 \pm \omega_0$ to shift by -90° (assuming $\omega_0 < \omega_{\rm S}/8$).

To analyze the effect of this phase shift, Fig. 8(e) shows a plot of the change in the phase spectrum for y_{23} compared to that for y_{12}. First, the phase at ω_0 does not change, and the phase at $\omega_{\rm S}/4 \pm \omega_0$ changes by -90°, as mentioned above. Second, at $-\omega_{\rm S}/4 \pm \omega_0$, the phase must change by $+90^\circ$ because the phase has to be antisymmetric to end up with real signals.

Next, to implement chopping as in Fig. 6, y_{12} and y_{23} are each multiplied by $(-1)^{\rm n} = \pm 1$. This operation is equivalent to multiplying by $\cos(\omega_{\rm S} t/2 + \pi/4)$, which shifts each spectrum by $\omega_{\rm S}/2$ and introduces a constant phase shift that can be ignored because it is the same in the y_{12} and y_{23} cases. Then, the chopped and nonchopped signals are multiplied by each other. In the y_{23} case, terms with a $+90^\circ$ phase shift are multiplied by terms with a -90° phase shift. The result appears at dc with the same magnitude but the opposite polarity that arises in the y_{12} case. Therefore, if the model outputs in the y_{12} and y_{23} cases are added, the dc terms cancel, avoiding the interference that appears when only y_{12} is used. This analysis ignores the z^{-1} delay elements at the output of the choppers in the timing detectors. (For example, Fig. 6 shows one of these delay elements for the y_{12} case.) However, these filters do not change the above result because they cause the same phase shift in both the y_{12} and y_{23} cases.

Fig. 9 shows a block diagram of the sample-time error detector for channel 2. Signals y_1–y_3 are the outputs after the offset and gain calibration system. Inside the proposed channel-2 timing detector, samples from channel 2 are added with the samples from channels 1 and 3 separately to produce y_{12} and y_{23}. These signals are then multiplied with chopped and delayed versions of themselves and scaled by $\mu_{\rm t}$ to produce

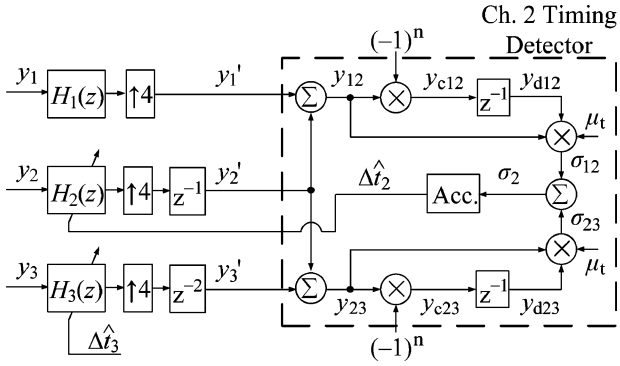

Fig. 9. Block diagram of sample-time error detector for channel 2.

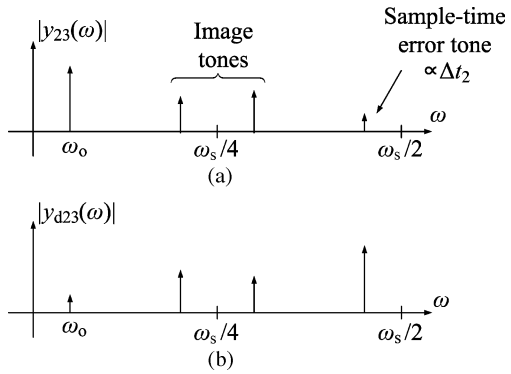

Fig. 10. Magnitude spectra with $\Delta t_2 \neq 0$ for (a) y_{23} and (b) $y_{\rm d23}$.

σ_{12} and σ_{23}. These signals are summed to produce σ_2. Then σ_2 is the input to this accumulator, which generates an estimate of the timing error in channel 2 without interference from the image tones at $\omega_{\rm S}/4 \pm \omega_0$ and this estimate Δt_2 controls the fractional delay of filter H_2.

Fig. 10 shows the magnitude spectra of y_{23} and $y_{\rm d23}$ with nonzero sample-time error Δt_2. As in the spectra for y_{12} in Fig. 7, nonzero Δt_2 causes a sample-time error tone to appear at $\omega_{\rm S}/2 - \omega_0$ in the y_{23} output, as shown in Fig. 10(a). Image tones appear at $\omega_{\rm S}/4 \pm \omega_0$ as well. Signal y_{23} is processed the same way as y_{12}, by chopping and delaying to produce $y_{\rm d23}$. Fig. 10(b) shows the result. Then, y_{23} and $y_{\rm d23}$ are multiplied. The products of the input and error tones at ω_0 and $\omega_{\rm S}/2 - \omega_0$ in these two signals produce a nonzero dc term with the same polarity as in the case of processing the same frequencies in y_{12}. Thus, when this dc term is added to the corresponding dc term from y_{12}, the two dc terms reinforce each other. On the other hand, the products of the image tones in y_{23} and $y_{\rm d23}$ produce a dc term that is equal in magnitude but opposite in polarity to the dc term that is generated from the image tones in channels 1 and 2. Therefore, when this dc term is added to the corresponding dc term from y_{12}, the two terms cancel, eliminating the interference from the image tones and leaving a dc term from the processing of the input and error tones that is approximately proportional to the sample time error, as desired. The elimination of the interference from the image tones is analyzed mathematically in the Appendix.

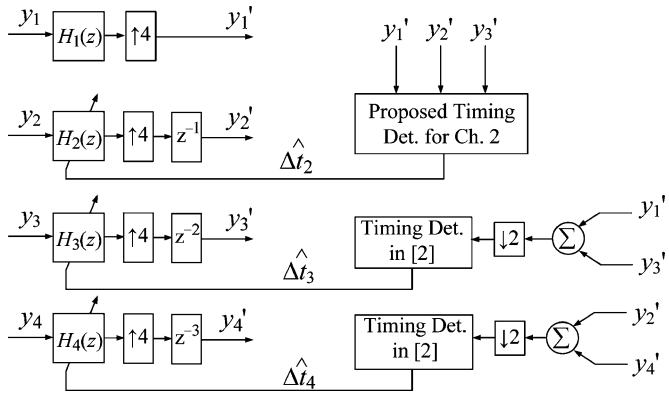

Fig. 11. Sample-time detector block diagram for a four-channel ADC.

Fig. 11 shows the sample-time detector block diagram for a four-channel ADC. Signals $y_1'-y_4'$ are outputs of the four channels after upsampling and proper time alignment. The sample-time error in channel 2 is calibrated using the timing detector shown in Fig. 9, which requires samples from the two adjacent channels as references. The sample-time errors in channels 3 and 4 are calibrated using Jamal's previous work [2]. That calibration technique relies on the fact that the ideal sampling time of the calibrated channel lays exactly half way between the sampling times of the reference channel. Therefore, channel 3 can be calibrated by using channel 1 as the reference. Likewise, sample-time error in channel 4 is calibrated with the same principle except using channel 2 as a reference. Since the calibration of channel 3 depends only on channel 1, $\Delta\hat{t}_3$ converges first (after about 50,000 samples). Then since the calibration of channel 2 depends on both channels 1 and 3, $\Delta\hat{t}_2$ converges next (after about 25,000 additional samples). Finally, $\Delta\hat{t}_4$ converges last (after about another 25,000 samples) because it depends on samples from channel 2. All together, about 100,000 samples are required, which is about twice as many as in the two-channel case because the convergences are staggered as described above. In steady state, each accumulator output produces an estimate of the sample-time error in the corresponding channel. This sample-time calibration technique assumes that the gain mismatches between channels have already been removed before the sample-time calibration is done.

Since the fractional delay FIR filters H_2–H_4 operate at the channel sampling rate, they operate without aliasing for input frequencies below half of the channel sampling rate. Since each channel samples only at a quarter of the overall output rate, this sample-time error-detection scheme works for an input with frequency content below $\omega_S/8$. Once the timing errors have been detected, Fig. 12 shows that four additional filters F_1–F_4 are used to produce the overall ADC output for inputs up to $\omega_S/2$. The inputs y_1–y_4 are the channel outputs after interchannel memory, offset and gain errors have been removed. Each of these inputs is upsampled by a factor of 4 and filtered by an FIR filter [$F_1(z)$, $F_2(z)$, $F_3(z)$, or $F_4(z)$] whose transfer function depends on all the sample-time errors [19]. To understand the design of these filters, assume the input is a complex exponential $e^{j\omega_o t}$ for simplicity and the Δt_i values are known. Then, the filters have to satisfy the following conditions [20]:

$$F_1(\omega_o)+F_2(\omega_o)e^{j\omega_o(T+\Delta t_2)}+F_3(\omega_o)e^{j\omega_o(2T+\Delta t_3)}+F_4(\omega_o)e^{j\omega_o(3T+\Delta t_4)}=4 \tag{1}$$

$$F_1\left(\frac{\omega_s}{4}+\omega_o\right)+F_2\left(\frac{\omega_s}{4}+\omega_o\right)e^{j\omega_o(T+\Delta t_2)}+F_3\left(\frac{\omega_s}{4}+\omega_o\right)e^{j\omega_o(2T+\Delta t_3)}+F_4\left(\frac{\omega_s}{4}+\omega_o\right)e^{j\omega_o(3T+\Delta t_4)}=0 \tag{2}$$

$$F_1\left(-\frac{\omega_s}{4}+\omega_o\right)+F_2\left(-\frac{\omega_s}{4}+\omega_o\right)e^{j\omega_o(T+\Delta t_2)}+F_3\left(-\frac{\omega_s}{4}+\omega_o\right)e^{j\omega_o(2T+\Delta t_3)}+F_4\left(-\frac{\omega_s}{4}+\omega_o\right)e^{j\omega_o(3T+\Delta t_4)}=0 \tag{3}$$

$$F_1\left(-\frac{\omega_s}{2}+\omega_o\right)+F_2\left(-\frac{\omega_s}{2}+\omega_o\right)e^{j\omega_o(T+\Delta t_2)}+F_3\left(-\frac{\omega_s}{2}+\omega_o\right)e^{j\omega_o(2T+\Delta t_3)}+F_4\left(-\frac{\omega_s}{2}+\omega_o\right)e^{j\omega_o(3T+\Delta t_4)}=0. \tag{4}$$

With nonzero sample-time errors, (1) indicates that the filter outputs from all four channels must sum up to 4 at ω_0 while (2)–(4) require the filter outputs to add up to zero at the image frequencies $\omega_S/4+\omega_0$, $-\omega_S/4+\omega_0$ and $-\omega_S/2+\omega_0$, respectively. Since each channel is down sampled by a factor of 4 (that is, it operates on only 1/4 of the overall samples), the sum in (1) is equal to 4 to compensate for the effect from down sampling. These requirements force the overall output spectrum to contain a nonzero component only at the input frequency ω_0 while eliminating all the images even with nonzero sample-time errors. An exact solution of these equations leads to complicated closed-form expressions for these filters [21]. Under the assumption of small sample-time errors (i.e., $\Delta t_i \ll T$, where $2 \le i \le 4$), however, an approximate closed-form solution of these equations is [21]

$$F_1(\omega)\approx 1-\frac{\pi}{4T}\{(\Delta t_2-\Delta t_4)S_e(\omega)+j(\Delta t_2+\Delta t_4)S_o(\omega)+j\Delta t_3 S_e(\omega)S_o(\omega)\} \tag{5}$$

$$F_2(\omega)\approx\left[1-j\frac{\Delta t_2}{T}\Psi(\omega)-\frac{\pi}{4T}\{\Delta t_3 S_e(\omega)+j\Delta t_3 S_o(\omega)+j\Delta t_4 S_e(\omega)S_o(\omega)\}\right]e^{-j\omega T} \tag{6}$$

$$F_3(\omega)\approx\left[1-j\frac{\Delta t_3}{T}\Psi(\omega)-\frac{\pi}{4T}\{-(\Delta t_2-\Delta t_4)S_e(\omega)+j(\Delta t_2+\Delta t_4)S_o(\omega)\}\right]e^{-2j\omega T} \tag{7}$$

$$F_4(\omega)\approx\left[1-j\frac{\Delta t_4}{T}\Psi(\omega)-\frac{\pi}{4T}\{-\Delta t_3 S_e(\omega)+j\Delta t_3 S_o(\omega)+j\Delta t_2 S_e(\omega)S_o(\omega)\}\right]e^{-3j\omega T} \tag{8}$$

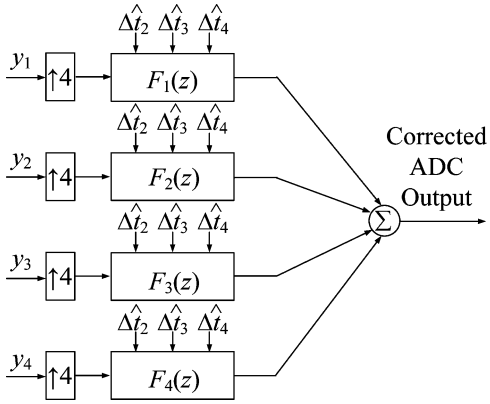

Fig. 12. Correction of sample-time errors and generation of the final ADC output with filters $F_1 - F_4$.

where $S_e(\omega)$, $S_o(\omega)$ and $\Psi(\omega)$ are given by

$$S_e(\omega) = \begin{cases} -1, & 0 \le |\omega| < \omega_s/4 \\ 1, & \omega_s/4 \le |\omega| < \omega_s/2 \end{cases} \tag{9}$$

$$S_o(\omega) = \mathrm{sgn}(\omega), \qquad -\omega_s/2 \le \omega < \omega_s/2 \tag{10}$$

$$\Psi(\omega) = \begin{cases} \omega T - \frac{\pi}{4}\mathrm{sgn}(\omega), & 0 \le |\omega| < \omega_s/4 \\ \omega T - \frac{3\pi}{4}\mathrm{sgn}(\omega), & \omega_s/4 \le |\omega| < \omega_s/2. \end{cases} \tag{11}$$

Equations (5)–(8) show that each $F_i(z)$ filter depends on all four timing errors. Therefore, a timing error in only one channel requires all four $F_i(z)$ filters in order to eliminate its effect in the final ADC output. Using the proposed correction technique shown in Fig. 12 with the correct timing-error estimates, the filters F_1–F_4 can correct timing errors for inputs up to $\omega_S/2$.

To perfectly compensate for sample-time errors, an infinite number of taps is required to implement the FIR filters. In practice, an approximation with a finite number of taps can be obtained by techniques such as windowing or minimizing the least-mean squared error [22]. Moreover, the number of taps can be optimized based on the ADC requirements [22], [23]. Measurements and simulations show that 21-tap and 61-tap FIR filters are adequate for H_2–H_4 and F_1–F_4, respectively. To realize the fractional delay filters H_2–H_4, a causal FIR approximation is used [18], which introduces a bulk delay of many samples in each filter. Therefore, $H_1(z) = z^{-10}$ is used in Figs. 6 and 11 to compensate for the delay through filters H_2–H_4 when all timing errors are zero. With 14-bit filter coefficients, the prototype is able to detect and correct timing errors satisfying $|\Delta t_i/T| < 0.15$ to 12-bit accuracy for frequencies as high as 90% of $\omega_S/2$ in the prototype. In general, more taps are needed for the filters to correct for larger timing errors or higher input frequencies [20].

C. Gain Calibration

Signal y_{12} in Fig. 9 is also used to sense gain mismatch between channels 1 and 2. Fig. 13 shows the output spectrum of y_{12} with a sinusoidal input and gain mismatch between the two channels. Gain mismatch causes amplitude modulation, generating a mismatch tone at $\omega_S/2 - \omega_0$ for an input at frequency ω_0. The amplitude of this tone is proportional to the gain mismatch

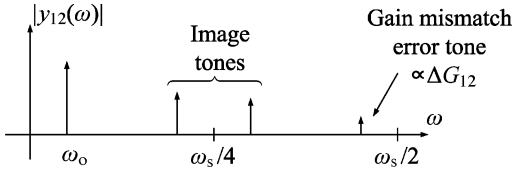

Fig. 13. Output spectrum of y_{12} with gain mismatch.

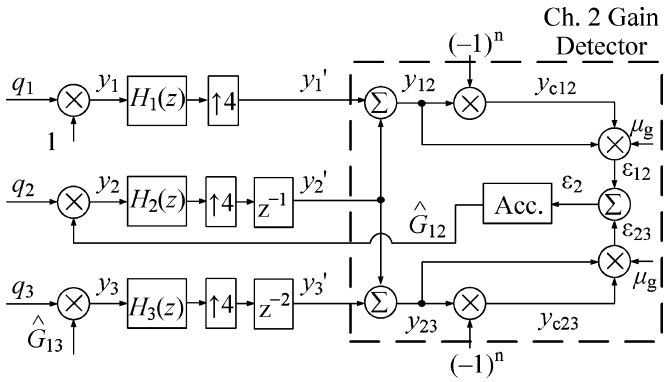

Fig. 14. Block diagram of gain calibration for channel 2.

between channels 1 and 2, ΔG_{12}, where $\Delta G_{12} = G_1 - G_2$ [24]. Also, image tones appear at $\omega_S/4 \pm \omega_0$ as a result of missing samples from channels 3 and 4. Nonzero gain mismatch alters the amplitude and phase of these image tones. The locations of the tones in Fig. 13 are the same as in Fig. 7. However, the error tone caused by gain mismatch in Fig. 13 is in phase with the input tone. In contrast, the error tone caused by timing error in Fig. 7 is orthogonal to the input tone. This difference allows gain mismatch and sample-time errors to be sensed and removed independently, as shown in [2]. Fig. 14 shows the first three channels and the gain calibration for channel 2. Signals q_1–q_3 are the outputs after the offset calibration block in Fig. 3. These signals are scaled to correct for gain mismatches. Channel 1 is the reference channel, so q_1 is multiplied by 1. q_2 is scaled by $\hat{G}_{12} = G_1/G_2$, which is an estimate of the ratio of the gains in channels 1 and 2. Similarly, q_3 is scaled by $\hat{G}_{13} = G_1/G_3$, which is an estimate of the ratio of the gains in channels 1 and 3. The outputs after gain mismatch correction are y_1–y_3, and these outputs are filtered for timing correction, upsampled, and delayed for time alignment to produce $y_1' - y_3'$ as shown in Figs. 9 and 14. Then, in Fig. 14, $y_1' - y_3'$ are input to the gain mismatch detector for channel 2, which finds $\hat{G}_{12}$. The detector that finds $\hat{G}_{13}$ is omitted for simplicity.

The gain detector here is exactly the same as the corresponding timing detector with one exception. The exception is that the unit delays after signals y_{c12} and y_{c23} have been removed. Those delays are used in the timing detector to avoid getting zero dc output at σ_2 even in the presence of nonzero sample-time error due to the orthogonality of the input and error tones. The delays are not used here because the input and error tones stemming from gain mismatch are in phase with each other. In practice, gain-mismatch and sample-time errors are likely to coexist in signals $y_1' - y_3'$. Fortunately, however, the gain detector here ignores the sample-time errors on average because the products of signals y_{12} and y_{c12} as well as y_{23}

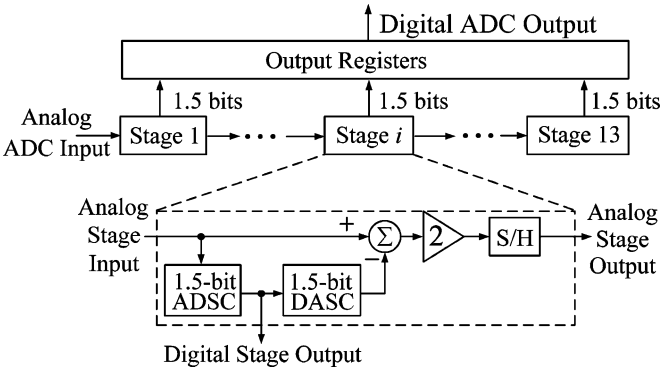

Fig. 15. Block diagram of the pipelined ADC in each channel.

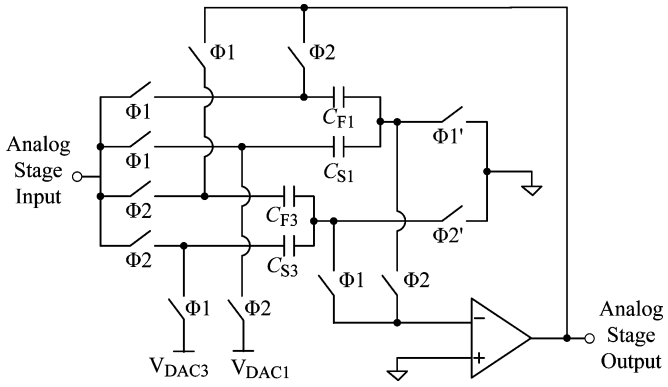

Fig. 16. Double-sampled MDAC for channels 1 and 3.

and y_{c23} related to sample-time errors average out to zero. The gain mismatch for channels 3 and 4 is calibrated using the gain detector described in [2] as in the approach of sample-time calibration shown in Fig. 11. Then, after gain mismatch is removed, the timing detectors described in the previous section correctly sense the timing errors. Convergence of the gain error estimates requires about 50,000 samples.

The gain mismatch and sample-time error calibration techniques described above are extendible to more than four channels in parallel. For example, for an eight-channel time-interleaved ADC, sample-time error can be calibrated using three of the timing detectors shown in Fig. 9 and four of the timing detectors in [2]. Sample-time errors in channels 2–4 would be sensed by timing detectors as in Fig. 9 with channels 1 and 3, channels 1 and 5, and channels 3 and 5 as references, respectively. Also, sample-time error in channels 5–8 would be sensed by timing detectors as in [2] with channels 1–4 as references, respectively. Also, a similar approach applies to gain mismatch calibration.

IV. Prototype Implementation

To demonstrate the digital calibration techniques described in the previous section, a prototype with two double-sampled pipelined ADCs was designed and fabricated in 0.35-μm n-well double-poly CMOS technology. Fig. 15 shows the block diagram of one of the pipelined channels. A 1.5-bit/stage architecture is used [25]–[27]. To implement four channels, two double-sampled pipelined ADCs were used on the prototype and each channel has 13 1.5-bit stages. Therefore, ideally, each channel gives almost 14-bit resolution. This large resolution was selected to allow thorough testing of the prototype. The measured results shown in Section V use outputs truncated to 11 bits for both calibration and evaluation of ADC performance.

Conventionally, a dedicated input sample-and-hold amplifier (SHA) is used in each channel to give the best performance at high input frequencies [28]. However, an input SHA would add noise and distortion to the analog input signal [29] and would require significant power dissipation. Fortunately, an input SHA is not always needed in practice [28]. Without an input SHA in each channel, a timing difference between the first-stage comparators and the first multiplying DAC (MDAC) generates an error. However, this error can be modeled as a frequency-dependent offset to the first-stage comparators. If this offset is less than the correction range provided by redundancy, it can be completely removed by standard digital correction of comparator offsets. The prototype includes an input SHA in each channel and an option to bypass them. The test results shown have the input SHAs bypassed.

Fig. 16 shows the single-ended version of the double-sampled MDAC used in the prototype. Two sets of sampling capacitors were used to implement double-sampling. Capacitors C_{S1} and C_{F1} are for channel 1, and C_{S3} and C_{F3} are for channel 3. $\Phi1$ and $\Phi2$ are standard non-overlapping clocks operating at the channel sampling rate. Since the opamp is used in a feedback loop to produce outputs on both phases, its input and output are never reset. While Fig. 16 shows a single-ended schematic for simplicity, all analog circuits on the prototype are fully differential. The opamp used in all the pipelined ADCs is a folded-cascode structure [30]. A switched-capacitor based comparator [31] is used for all of the stages and the comparator core in the pipelined ADC uses the architecture shown in [32].

A required functional block in the memory calibration system is the pseudo-random-number generator (RNG). Two RNGs are implemented on the chip and each uses a maximum-length shift-register structure with 41 stages in the shift register [33], generating a binary sequence that is white and uncorrelated with the input signal. With 41 stages, the resulting period is about 15 h, and the periodicity of the noise is well below any measurement frequency of interest.

Fig. 17 shows the die photograph of the prototype. It is fabricated in a 0.35-μm double-poly CMOS technology, and is approximately 5.2 mm $\times$ 5.2 mm. The total area is 27 mm^2, and the active area is about 13.9 mm^2. The two double-sampled pipelined ADCs together give four time-interleaved channels. The first three stages in each channel are calibrated for interchannel memory error. The channel outputs are calibrated for gain mismatch and timing errors. Offset mismatch between channels can be overcome in the background using the random chopper calibration technique described in [2]. For simplicity, however, offset calibration is done in the foreground here by digitizing and subtracting the mean of each channel. All of the digital calibration is done off chip in software. Table I shows estimates of the die area and power dissipation that would be required to implement each of the calibration blocks in both 0.35-μm and 0.13-μm CMOS. The F filters in Fig. 12 dominate the area and power dissipation requirements, causing the timing calibration to be impractical in 0.35-μm CMOS. However, the

TABLE I
ESTIMATED DIE AREAS AND POWER DISSIPATIONS OF CALIBRATION BLOCKS

	0.35 µm		0.13 µm	
	Area (mm^2)	Power (mW)	Area (mm^2)	Power (mW)
Gain Cal. & Timing Detection	5.0	320	0.7	42
The *F* filters	14.4	920	2.0	120
Memory Cal.	1.2	80	0.17	10
Calibration total	20.6	1320	2.9	170

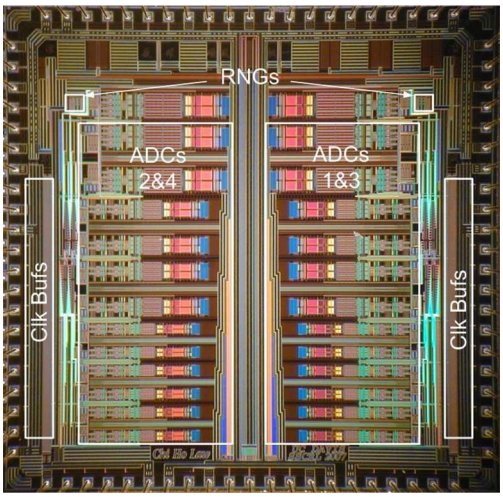

Fig. 17. Die photograph.

cost of interchannel memory calibration is less than the savings in the analog domain, where interchannel memory calibration enables the use of double sampling, reducing the analog power dissipation and area each by about a factor of two. Also, the estimated cost of timing calibration is reasonable in 0.13-μm CMOS.

V. EXPERIMENTAL RESULTS

Unless otherwise stated, for all tests, the input is a sinusoid with amplitude of 1.8 V peak-to-peak fully differential, the sampling rate is 160 Msamples/s, and the power supply is 3.3 V.

Fig. 18 shows ADC output spectra for an input frequency of 43.309 MHz. The sample rate is 160 Msamples/s, but the ADC output was down-sampled by a factor of 37 to reduce the number of output pins on chip. The downsampling allowed eight output pins to be time multiplexed to handle the four ADC channel outputs. Due to the downsampling, the fast Fourier transform (FFT) plots extend to half the sample rate divided by 37, or 2.2 MHz. Fig. 18(a) shows the output of the ADC without calibration. After aliasing, the 43.309-MHz input tone appears at 66 kHz. Gain and timing mismatches cause tones to appear at $f_s/4-f_0$, $f_s/4+f_0$ and $f_s/2-f_0$, which are located at 1.015, 1.147, and 2.096 MHz after aliasing, respectively. Without calibration, the SNDR is 31.2 dB and the SFDR is 32.3 dB. Fig. 18(b) shows the output spectrum with calibration only for gain mismatch between the channels, and the SNDR and SFDR have improved to 32.0 and 32.8 dB. The improvement here is small because timing mismatch causes the dominant error in this case.

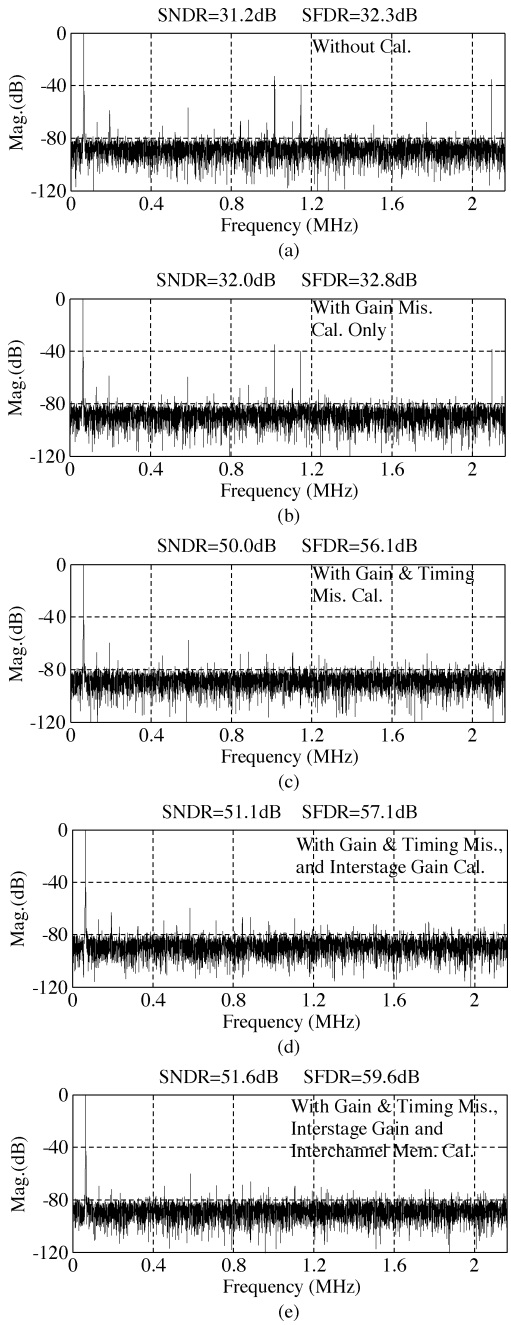

Fig. 18. ADC output spectra with $f_0 = 43.309$ MHz (a) without calibration, (b) with gain mismatch calibration only, (c) with gain and timing mismatch calibration, (d) with gain and timing mismatch calibration as well as interstage gain calibration and (e) with the aforementioned calibrations plus the interchannel memory calibration.

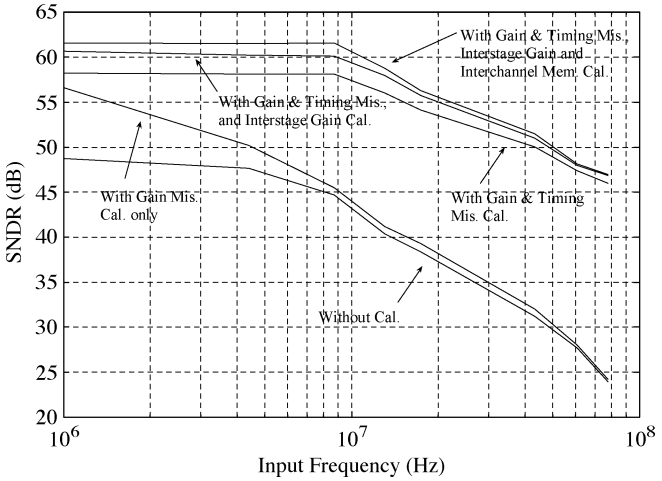

Fig. 19. SNDR versus input frequency f_0 ($f_S = 160$ MSamples/s).

Fig. 18(c) shows the output of the ADC with calibration for gain and timing mismatches. The gain and timing tones have decreased by more than 40 dB into the noise floor, and the ninth harmonic distortion becomes the largest undesired tone. The overall SNDR and SFDR here have improved by about 18 dB to 50.0 and 56.1 dB, respectively. Fig. 18(d) shows the output spectrum with calibration for not only channel gain and timing mismatches, but also for interstage gain errors. Interstage gain calibration improves the SNDR and SFDR by about 1 dB each. Fig. 18(e) shows the ADC output with all the aforementioned calibrations plus the interchannel memory calibration, which further improves the SNDR by about 0.5 dB and the SFDR by about 2.5 dB.

Fig. 19 shows five plots of SNDR versus input frequency. The bottom plot is without calibration except for the digital subtraction of channel offsets. The next plot up is with calibration for only gain mismatches. The gain error coefficients converge to values ranging from about 0.1% to 0.2%. At low input frequencies, gain mismatch is the dominant error, and a significant improvement is obtained by calibration for gain mismatch. However, the importance of timing mismatch increases with the input frequency, reducing the improvement. The third plot is with calibration for only gain and timing mismatches. The timing error estimates converge to values ranging from about 300 to 800 ps. (The timing errors are dominated by loose adjustment of the phases of four 40-MHz sampling clocks, one for each channel.) Compared to the bottom plot, the SNDR has improved by more than 13 dB to 58 dB for input frequencies up to 10 MHz. The fourth plot also includes interstage gain calibration. The low-frequency SNDR improves by about 2 dB. The top plot also includes interchannel memory calibration in addition to all the other calibrations previously mentioned. Memory calibration improves the maximum SNDR by about 1 dB, and the memory-effect coefficients converge to values ranging from about 0.1% to 0.2%. With all calibrations, the SNDR is at least 50 dB out to 50 MHz. This limit stems from sampling jitter, which is about 12 ps rms. These measurements use 21-tap and 61-tap FIR filters for H_2–H_4 and F_1–F_4, respectively. The number of taps needed in other applications is approximately proportional to the maximum timing error and inversely proportional to the jitter in the sampling clock.

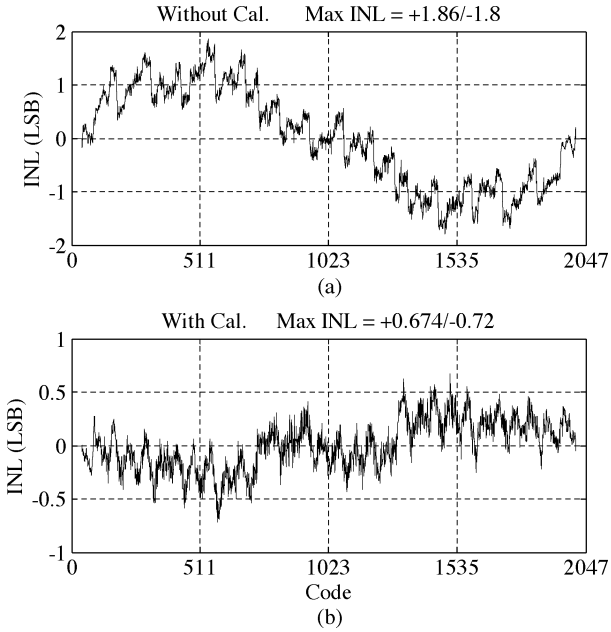

Fig. 20. INL versus code (a) without calibration and (b) with calibration ($f_S = 160$ MSamples/s, $f_0 = 70$ kHz).

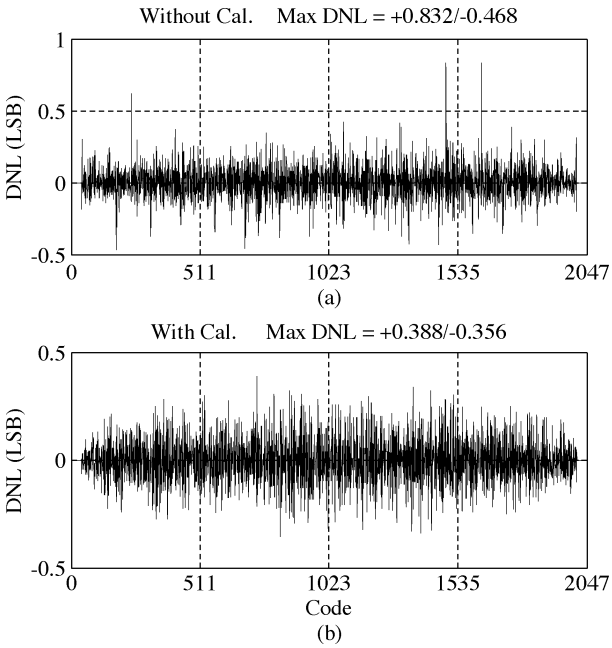

Fig. 21. DNL versus code (a) without calibration and (b) with calibration ($f_S = 160$ MSamples/s, $f_0 = 70$ kHz).

Fig. 20 shows the integral nonlinearity (INL) plots from code-density tests [34] on the ADC output, including all four interleaved channels. The input frequency is 70 kHz. Fig. 20(a) shows the plot of INL without calibration, and Fig. 20(b) shows the plot of INL with gain and timing mismatch, interstage gain and interchannel memory calibration. Without calibration, the peak INL is 1.86 least significant bits (LSB), and with calibration, the peak INL reduces to 0.72 LSB. Fig. 21(a) shows the plot of differential nonlinearity (DNL) without calibration, and

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

Process	0.35 μm 2P4M CMOS	
Resolution	11-bit	
Sampling Rate f_S	160 Msamples/s	
Active Area	13.3 mm²	
Full-Scale Input	1.8 Vp-p	
	Without Cal.	With Cal.
Power Diss. (Analog / Total)	540 / 594 mW	540 / 594 mW + External
Max. |INL| (f_o=8.7 MHz)	1.86 LSB	0.72 LSB
Max. |DNL| (f_o=8.7 MHz)	0.83 LSB	0.39 LSB
SNDR (f_o=8.7 MHz)	44.8 dB	61.6 dB
SFDR (f_o=8.7 MHz)	47.3 dB	78.6 dB
THD (f_o=8.7 MHz)	–57.9 dB	–77.8 dB
PSRR (f_{supply}=0.05 MHz)	63.0 dB	63.2 dB
CMRR (f_o=8.7 MHz)	69.1 dB	68.9 dB

Fig. 21(b) shows the plot of DNL with the same calibration as in Fig. 20(b). Without calibration, the peak DNL is 0.83 LSB, and with calibration, the peak DNL reduces to 0.39 LSB. Table II summarizes the measured performance.

VI. CONCLUSION

A four-channel time-interleaved ADC with calibration is described. It uses gain and timing calibration techniques that are extendable to any number of channels but are applied in the foreground because they limit the bandwidth during calibration. Also, the prototype demonstrates background interchannel memory calibration for the first time. This work potentially expands the range of applications in which time-interleaving is a viable solution to reach the high conversion rates and low power required in future ADC products.

APPENDIX

This Appendix analyzes the sample-time error detector for channel 2 shown in Fig. 9 for an arbitrary input and shows that the average accumulator input is proportional to the timing error in channel 2.

A. Processing y_{12}

An expression for the signal σ_{12} in Fig. 9 with sample-time error only in channel 2 is found below. Assume that channel 2 samples at a time $T+\Delta t_2$ after channel 1, so the sample-time error in channel 2 is Δt_2. Since the samples from each ADC channel are up-sampled by inserting three zero samples for each channel output sample, the signal y_{12} contains samples from channel 1 taken at times $m4T$, samples from channel 2 at times $m4T+T+\Delta t_2$, and two zero samples at times $m4T+2T$ and $m4T+3T$. The samples $y_{12}[n]$ can be expressed as

$$y_{12}[n]=\left(y(t)\big|_{t=nT+\left(\frac{(1-(-1)^n)}{4}+\frac{\sin(n\pi/2)}{2}\right)\Delta t_2}\right)\cdot\underbrace{\frac{(1+\sin(n\pi/2)+\cos(n\pi/2))}{2}}_{\text{Periodic term}}. \quad \text{(A1)}$$

The first term in (A1) generates samples of $y(t)$ at the times: $0, T+\Delta t_2, 2T, 3T, 4T, 5T+\Delta t_2, 6T, 7T, \ldots$ So this equation models sample-time error only in channel 2. The second term in (A1) is a periodic "1, 1, 0, 0" signal. The product of these two terms is y_{12}, which is nonzero only when $n=0,1,4,5,8,9,\ldots$, that is, at the sample times associated with channels 1 and 2.

Using a first-order Taylor series expansion for $y(t)$ in (A1) assuming that $|\Delta t_2/T| \ll 1$, y_{12} can be expressed as

$$y_{12}[n]=\left(y(nT)+\Delta t_2\left(\frac{(1-(-1)^n)}{4}+\frac{\sin(n\pi/2)}{2}\right)\times\frac{dy(nT)}{dt}\right)\cdot\left(\frac{1+\sin(n\pi/2)+\cos(n\pi/2)}{2}\right). \quad \text{(A2)}$$

Signal y_{12} is chopped by $(-1)^n$ and then delayed by one sample to produce y_{d12}

$$\begin{aligned}y_{d12}[n]&=y_{C12}[n-1]\\&=(-1)^{(n-1)}y_{12}[n-1]=(-1)^{(n-1)}\\&\quad\times\left(y\left((n-1)T\right)+\Delta t_2\left(\frac{\left(1-(-1)^{(n-1)}\right)}{4}+\frac{\sin\left((n-1)\pi/2\right)}{2}\right)\times\frac{dy\left((n-1)T\right)}{dt}\right)\\&\quad\cdot\left(\frac{1+\sin\left((n-1)\pi/2\right)+\cos\left((n-1)\pi/2\right)}{2}\right). \quad \text{(A3)}\end{aligned}$$

Then y_{12} and y_{d12} are multiplied and the resulting product can be written as (after significant manipulation)

$$y_{12}[n]y_{d12}[n]=\left(y(nT)+\Delta t_2\frac{dy(nT)}{dt}\right)\cdot\left(y\left((n-1)T\right)\right)\cdot\left(\frac{(\sin(n\pi/2))^2+\sin(n\pi/2)}{2}\right). \quad \text{(A4)}$$

The last term in (A4) is a periodic "0, 1, 0, 0" signal, which has frequency components at dc, $f_S/4$ and $f_S/2$.

Assume the input $y(t)$ is bandlimited to $f_S/8$. In that case, the only dc component in (A4) is the dc component of the last term on the right (which is 1/4) times the dc component of the product of the first two terms. Therefore, the dc term in σ_{12}, which is the dc term in (A4) scaled by μ_t, is

$$\begin{aligned}\overline{\sigma_{12}}&=\mu_t\cdot\overline{y_{12}[n]y_{d12}[n]}\\&=\mu_t\cdot\frac{1}{4}\left(\overline{y(nT)y\left((n-1)T\right)}+\Delta t_2\overline{\frac{dy(nT)}{dt}y\left((n-1)T\right)}\right). \quad \text{(A5)}\end{aligned}$$

This dc term is of interest here because σ_{12} is fed into an accumulator, which is in a negative feedback loop. Therefore, in steady state, the dc component of the accumulator input is driven to zero by the feedback. In (A5), the term proportional to Δt_2 is the desired term because it allows the timing error to be sensed;

however, the term related to $\overline{y(nT)y((n-1)T)}$ is undesired because it does not contribute to sensing the timing error.

B. Processing y_{23}

Following a similar analysis, the processing of y_{23} shown in Fig. 9 generates a product of y_{23} and $y_{\mathrm{d}23}$ that can be simplified to

$$\begin{aligned} y_{23}[n]y_{d23}[n] = &(y(nT)) \\ &\cdot(-1)\left(y((n-1)T)+\Delta t_2\frac{dy((n-1)T)}{dt}\right) \\ &\cdot\left(\frac{(\cos(n\pi/2))^2-\cos(n\pi/2)}{2}\right). \end{aligned} \tag{A6}$$

Assuming the input $y(t)$ is bandlimited to $f_{\mathrm{S}}/8$, the dc term in σ_{23}, which is the dc term in (A6) scaled by μ_t, can be shown to be

$$\begin{aligned} \overline{\sigma_{23}} &= \mu_t\cdot\overline{y_{23}[n]y_{d23}[n]} \\ &= \mu_t\cdot\left(\frac{-1}{4}\right)\left(\overline{y(nT)y((n-1)T)}\right. \\ &\qquad \left.+\Delta t_2\overline{\frac{dy((n-1)T)}{dt}y(nT)}\right). \end{aligned} \tag{A7}$$

This equation shows that the dc term generated from the processing of y_{23} contains a desired term proportional to Δt_2 and an undesired term that contains $\overline{y(nT)y((n-1)T)}$.

C. Accumulator Input

The signals σ_{12} and σ_{23} are summed to produce the accumulator input σ_2 in Fig. 9. Using the results in this Appendix, the dc component of σ_2 is

$$\begin{aligned} \overline{\sigma_2} &= \overline{\sigma_{12}+\sigma_{23}} \\ &= \mu_t\cdot\left(\overline{y_{12}[n]y_{d12}[n]}+\overline{y_{23}[n]y_{d23}[n]}\right) \\ &= \mu_t\cdot\frac{\Delta t_2}{4}\left(\overline{\frac{dy(nT)}{dt}y((n-1)T)}\right. \\ &\qquad \left.-\overline{\frac{dy((n-1)T)}{dt}y(nT)}\right). \end{aligned} \tag{A8}$$

This equation shows that the dc component of σ_2 is proportional to Δt_2. Summing (A5) and (A6) eliminates the undesired terms. Therefore, the timing detector for channel 2 senses Δt_2. Due to negative feedback, the average accumulator input must be zero in steady state. When that occurs, the filter $H_2(z)$ has adjusted to correct for the sample-time error Δt_2 in channel 2.

ACKNOWLEDGMENT

The authors are grateful to N. Chang, B. C. Levy, D. Stoops, and H. Wang for technical discussions, as well as to the reviewers for helpful technical feedback.

REFERENCES

[1] W. C. Black, Jr. and D. A. Hodges, "Time interleaved converter arrays," *IEEE J. Solid-State Circuits*, vol. SC-15, no. 6, pp. 1022–1029, Dec. 1980.

[2] S. M. Jamal, D. Fu, N. C. J. Chang, P. J. Hurst, and S. H. Lewis, "A 10-b 120-Msample/s time-interleaved analog-to-digital converter with digital background calibration," *IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1618–1627, Dec. 2002.

[3] 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, no. 12, pp. 1904–1911, Dec. 1998.

[4] Y. C. Jenq, "Digital spectra of nonuniformly sampled signals: Fundamentals and high-speed waveform digitizers," *IEEE Trans. Instrum. Meas.*, vol. 37, no. 2, pp. 245–251, Jun. 1988.

[5] C. S. G. Conroy, D. W. Cline, and P. R. Gray, "An 8-b 85-MS/s parallel pipeline A/D converter in 1- μm CMOS," *IEEE J. Solid-State Circuits*, vol. 28, no. 4, pp. 447–454, Apr. 1993.

[6] L. Sumanen, M. Waltari, and K. A. I. Halonen, "A 10-bit 200-MS/s CMOS parallel pipeline A/D converter," *IEEE J. Solid-State Circuits*, vol. 36, no. 7, pp. 1048–1055, Jul. 2001.

[7] K. C. Dyer, D. Fu, S. H. Lewis, and P. J. Hurst, "An analog background calibration technique for time-interleaved analog-to-digital converters," *IEEE J. Solid-State Circuits*, vol. 33, no. 12, pp. 1912–1919, Dec. 1998.

[8] H. Jin and E. K. F. Lee, "A digital-background calibration technique for minimizing timing-error effects in time-interleaved ADCs," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 47, no. 7, pp. 603–613, Jul. 2000.

[9] K. Y. Kim, N. Kusayanagi, and A. A. Abidi, "A 10-b, 100-MS/s CMOS A/D converter," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 302–311, Mar. 1997.

[10] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara, and K. Kobayashi, "Explicit analysis of channel mismatch effects in time-interleaved ADC systems," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 48, no. 3, pp. 261–271, Mar. 2001.

[11] S. Limotyrakis, S. D. Kulchycki, D. K. Su, and B. A. Wooley, "A 150-MS/s 8-b 71-mW CMOS time-interleaved ADC," *IEEE J. Solid-State Circuits*, vol. 40, no. 5, pp. 1057–1067, May 2005.

[12] W. Bright, "8b 75 MSample/s 70 mW parallel pipelined ADC incorporating double sampling," in *ISSCC Dig. Tech. Papers*, Feb. 1998, pp. 146–147.

[13] L. Wu and W. C. Black, Jr., "A low-jitter skew-calibrated multi-phase clock generator for time-interleaved applications," in *ISSCC Dig. Tech. Papers*, Feb. 2001, pp. 396–397.

[14] C.-Y. Wang and J.-T. Wu, "A background timing-skew calibration technique for time-interleaved analog-to-digital converters," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 4, pp. 299–303, Apr. 2006.

[15] D. Camarero, K. B. Kalaia, J.-F. Naviner, and P. Loumeau, "Mixed-signal clock-skew calibration technique for time-interleaved ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 11, pp. 3676–3687, Dec. 2008.

[16] J. Keane, P. J. Hurst, and S. H. Lewis, "Digital background calibration for memory effects in pipelined analog-to-digital converters," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 3, pp. 511–525, Mar. 2006.

[17] D. Garrity, D. LoCascio, C. Cavanagh, M. N. Kabir, and C. Guenther, "A single analog-to-digital converter that converts two separate channels (I and Q) in a broadband radio receiver," *IEEE J. Solid-State Circuits*, vol. 43, no. 6, pp. 1458–1469, Jun. 2008.

[18] T. I. Laakso, V. Välimäki, M. Karjalainen, and U. K. Laine, "Splitting the unit delay—Tools for fractional delay filter design," *IEEE Signal Processing Mag.*, vol. 13, no. 1, pp. 30–60, Jan. 1996.

[19] R. S. Prendergast, B. C. Levy, and P. J. Hurst, "Reconstruction of bandlimited periodic nonuniformly sampled signals through multirate filter banks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 8, pp. 1612–1622, Aug. 2004.

[20] S. M. Jamal, D. Fu, M. P. Singh, P. J. Hurst, and S. H. Lewis, "Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 1, pp. 130–139, Jan. 2004.

[21] S. Huang and B. C. Levy, "Blind calibration of timing offsets for four-channel time-interleaved ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 4, pp. 863–876, Apr. 2007.

[22] G. D. Cain, N. P. Murphy, and A. Tarczynski, "Evaluation of several variable FIR fractional-sample delay filters," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Apr. 1994, vol. 3, pp. 621–624.
[23] C. S. Burrus, A. W. Soewito, and R. A. Gopinath, "Least squared error FIR filter design with transition bands," *IEEE Trans. Signal Process.*, vol. 40, no. 6, pp. 1327–1340, Jun. 1992.
[24] C. S. G. Conroy, "A high-speed parallel pipeline A/D converter technique in CMOS," Ph.D. dissertation, Dept. EECS, Univ. California, Berkeley, Feb. 1994.
[25] G. Jusuf, "A 1-bit/cycle algorithmic analog-to-digital converter without high-precision comparators," Univ. California, Berkeley, Memo. UCB/ERL M90/69, Aug. 1990.
[26] S. H. Lewis, H. S. Fetterman, G. F. Gross, Jr., R. Ramachandran, and T. R. Viswanathan, "A 10-b 20-Msample/s analog-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 27, no. 3, pp. 351–358, Mar. 1992.
[27] 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, no. 7, pp. 957–965, Jul. 1992.
[28] 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, no. 3, pp. 166–172, Mar. 1995.
[29] D.-Y. Chang, "Design techniques for a pipelined ADC without using a front-end sample-and-hold amplifier," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 11, pp. 2123–2132, Nov. 2004.
[30] 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 communications applications," *IEEE J. Solid-State Circuits*, vol. SC-18, no. 6, pp. 652–664, Dec. 1983.
[31] X. Wang, P. J. Hurst, and S. H. Lewis, "A 12-Bit 20-Msample/s pipelined ADC with nested digital background calibration," *IEEE J. Solid-State Circuits*, vol. 39, no. 11, pp. 1799–1808, Nov. 2004.
[32] K. Nagaraj, H. Fetterman, J. Anidjar, S. Lewis, and R. Renninger, "A 250-mW, 8-b, 52-Msamples/s parallel-pipelined A/D converter with reduced number of amplifiers," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 312–320, Mar. 1997.
[33] R. C. Dixon, *Spread Spectrum Systems*. New York: Wiley, 1976.
[34] J. Doernberg, H.-S. Lee, and D. A. Hodges, "Full-speed testing of A/D converters," *IEEE J. Solid-State Circuits*, vol. SC-19, no. 6, pp. 820–827, Dec. 1984.

Chi Ho Law (S'02) received the B.S., M.S., and PhD degrees from the University of California, Davis, in 2002, 2005, and 2009, respectively, all in electrical and computer engineering.

Since 2009, he has been a Staff Design Engineer with Exar Corporation, Fremont, CA. His research interests include smart temperature sensor design and analog and mixed-signal circuit design in both data and 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, Berkeley, in 1977, 1979, and 1983, respectively.

From 1983 to 1984, he was with the University of California, Berkeley, as a Lecturer, teaching integrated-circuit design courses and working on an MOS delta-sigma modulator. In 1984, he joined Silicon Systems Inc., Nevada City, CA, 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, University of California, 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 has served on the program committees for the Symposium on VLSI Circuits and the International Solid-State Circuits Conference. He is a coauthor of a textbook on analog integrated-circuit design. He is also active as a consultant to industry.

Prof. Hurst served for five years as an associate editor for the IEEE JOURNAL OF SOLID-STATE CIRCUITS. He is now a member of the administrative committee of the IEEE Solid-State Circuits Society.

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, 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, Davis, where he is now a Professor. He is a coauthor of a college textbook on analog integrated circuits, and his research interests include data conversion, signal processing, and analog circuit design.