A 12-GS/s 81-mW 5-bit Time-Interleaved Flash ADC With Background Timing Skew Calibration

Manar El-Chammas, *Member, IEEE*, and Boris Murmann, *Senior Member, IEEE*

***Abstract*—This paper presents a 12-GS/s 5-bit time-interleaved flash ADC realized in 65-nm CMOS. To improve the dynamic performance at high input frequencies, a statistics-based background calibration scheme for timing skew is employed. The timing skew is detected in the digital domain through a correlation-based algorithm and minimized by adjusting digitally controlled delay lines. In order to minimize power consumption, we employ near minimum size comparators, whose offset is reduced through foreground calibrated trim-DAC circuitry. With the timing calibration activated, the skew-related impairments are reduced by 12 dB at high input frequencies, resulting in an SNDR of 25.1 dB near Nyquist. The prototype IC consumes 81 mW from a 1.1 V supply, yielding a figure-of-merit of 0.35 pJ/conversion-step at low input frequencies, and 0.46 pJ/conversion-step for inputs near Nyquist.**

***Index Terms*—A/D conversion, calibration, flash ADCs, time-interleaving.**

I. Introduction

As THE baud rate of wireline communication systems continues to increase, increasingly complex equalization blocks are required to meet the target bit-error rate. This has resulted in a trend towards digitally-equalized serial links, requiring the use of high-speed analog-to-digital converters to digitize the incoming signal.

The specifications for the required ADC depend on the application, and can range from a sample rate and bit resolution of 1 GS/s and 11-bits [1] to over 10 GS/s and 4-bits [2]. Even in today's fine-line technology, such speed-resolution products are realizable with acceptable power efficiency only by exploiting the time-domain parallelism of interleaved architectures [3]. Unfortunately, time-interleaved ADCs suffer from time-varying errors related to mismatch between the individual conversion channels. The primary issue in the performance range considered here is timing skew, which results in non-uniform sampling of the input. The impact of timing skew increases with input frequency, and therefore tends to overshadow the effect of other non-idealities for broadband inputs.

The converter discussed in this paper overcomes the effects of timing skew through the use of a statistics-based background calibration algorithm that drives digitally controlled delay lines toward minimum timing skew between the interleaved channels. In addition, our design employs a foreground calibration scheme to address offsets from near-minimum size comparators used in the flash sub-ADCs. This reduces power in the active circuitry, as well as in the resistive reference ladder, which can be designed for higher impedance due to reduced kickback charge. Together with 8-fold interleaving, which results in a relatively low per-channel conversion rate, this yields an overall power dissipation that meets the stringent requirements of high-speed serial link applications.

This paper is structured as follows. Section II describes the ADC architecture used in the prototype IC. Section III discusses the effect of time-varying errors in interleaved ADCs and presents various methods that can be used to compensate for these errors. Section IV introduces the proposed statistics-based background calibration algorithm and the required support circuitry for skew estimation and adjustment. Section V covers critical ADC sub-circuits along with the offset calibration scheme used to minimize comparator offset. Measurement results for the static and dynamic performance of the ADC, as well as calibration convergence results are shown in Section VI, followed by a conclusion in Section VII.

II. ADC Architecture Overview

Fig. 1 shows the block diagram of the 5-bit prototype ADC. The design uses an interleaving factor of eight, resulting in a sub-ADC sample rate of 1.5 GS/s. The phase generator creates the eight required phases, each of which are fed into a digitally controlled delay line that allows for timing skew adjustments. A redundant single-comparator conversion channel (labeled CAL) controlled by an independent clock is added to the architecture for timing skew calibration. This channel samples the input at a reduced rate and drives the off-chip timing skew calibration engine, discussed later in more detail. The output of the time-interleaved ADC array is decimated by a factor of 81 to ease the I/O bandwidth requirements during testing.

III. Time-Interleaved ADCs

A time-interleaved ADC is an architecture that cycles through a set of N sub-ADCs [3], such that the aggregate throughput is N times the sample rate of the individual sub-ADCs (see Fig. 2). The sampling period of the time-interleaved ADC and the N sub-ADCs are T_s and $N \cdot T_s$, respectively. The sampling edges of two consecutive clocks are offset by T_s such that the input signal is uniformly sampled. Thus, the output of the i^{th} sub-ADC is $y_i[n]$, where

$$y_i[n] = x\left(t - (nN + i)\,T_s\right). \tag{1}$$

Manuscript received August 24, 2010; revised November 15, 2010; accepted December 15, 2010. Date of publication March 03, 2011; date of current version March 25, 2011. This paper was approved by Guest Editor Ajith Amerasekera. This work was supported by the Stanford Initiative for Rethinking Analog Design (RAD). The fabrication of this chip was made possible by the TSMC University Shuttle Program.

M. El-Chammas was with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA, and is now with Texas Instruments Inc., Dallas, TX 75243 USA (e-mail: manar.chammas@gmail.com).

B. Murmann is with the Department of Electrical Engineering, Stanford University, Stanford, CA 94305 USA (e-mail: murmann@stanford.edu).

Digital Object Identifier 10.1109/JSSC.2011.2108125

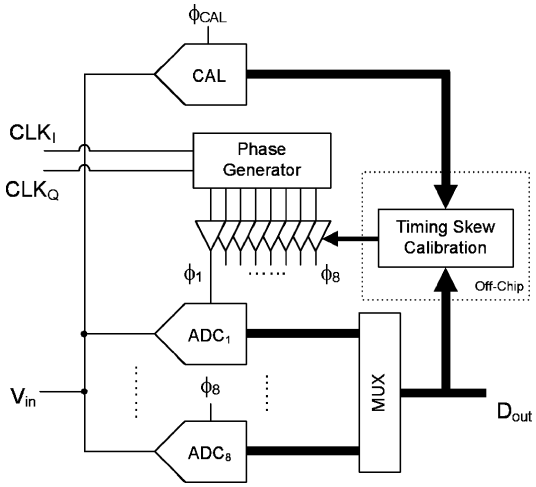

Fig. 1. ADC block diagram. The actual input and clock signal path is differential.

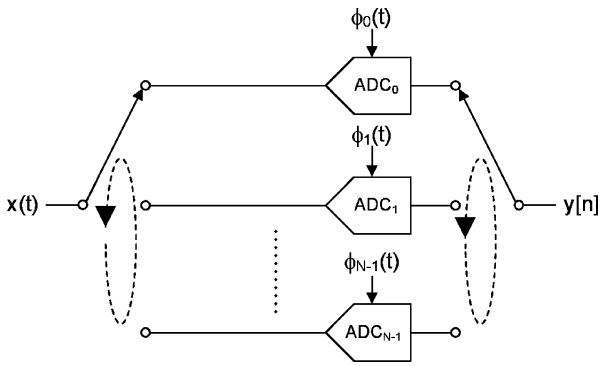

Fig. 2. Time-interleaved ADC.

Multiplexing the sub-ADC outputs results in

$$y[n] = x\left(nT_s\right) \tag{2}$$

as the time-interleaved ADC output.

However, variations in the signal and clock paths result in time-varying errors, the most prominent of which are due to gain, offset, and timing skew, as shown in Fig. 3. This modifies (1) into

$$y_i[n] = G_i x\left(t - (nN + i)\,T_s - \tau_i\right) + o_i \tag{3}$$

where G_i is the sub-ADC gain, o_i the sub-ADC offset, and τ_i the sub-ADC timing skew. Thus, the time-interleaved output with mismatches present in each channel does not reduce to (2) as a result of these errors. Acquisition bandwidth variations between the channels also result in time-varying errors. However, in the presented design, the effect of this mismatch is reduced by using a sufficiently large tracking bandwidth when compared to the input frequency [4].

A. Impact of Timing Skew

Of the aforementioned time-varying errors, the impact of timing skew is dominant for the given design target, and is the focus of this paper. Timing skew results in the non-uniform sampling of the input signal, which creates voltage sampling

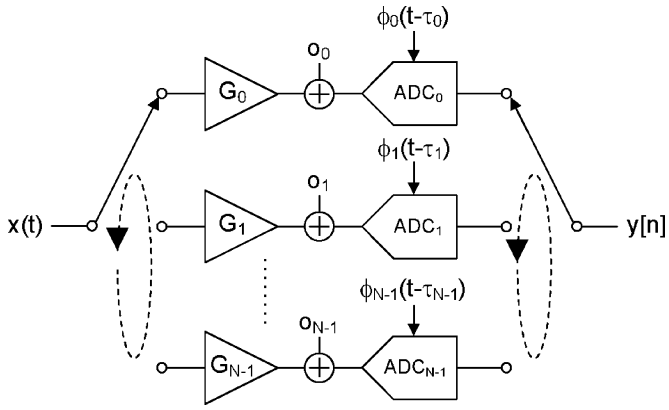

Fig. 3. Mismatch model for a time-interleaved ADC.

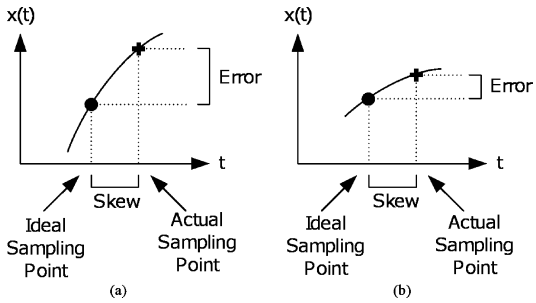

Fig. 4. Sampling error as a result of timing skew for (a) high frequency signals and (b) low frequency signals.

errors, as illustrated in Fig. 4. The magnitude of this sampling error changes as a function of the frequency of the input signal. Higher frequency input signals have larger changes for a given timing skew, resulting in a larger sampling error. Lower frequency input signals have smaller changes for a given timing skew, and the resulting sampling error is not as large. This is not the case for both gain and offset mismatch. Quantitative relationships between timing skew and ADC performance can be derived for wide-sense stationary and wide-sense cyclostationary signals [5] and relate the autocorrelation of the input signal to the ADC's *SNR*. A simple estimate for the maximum tolerable skew is obtained when the input signal is assumed to be a sinusoid with frequency f, and when the skew induced error is bounded in magnitude to the quantization error of a B-bit converter. This yields [5]

$$\sigma_\tau^2 \leq \left(\frac{N}{N-1}\right) \cdot \left(\frac{2}{3 \cdot 2^{2B}}\right) \cdot \left(\frac{1}{(2\pi f)^2}\right) \tag{4}$$

where σ_τ is the standard deviation of the timing skew among the channels and N is the interleaving factor. Assuming that the input is a wideband signal with a uniform power spectral density in the interval $0 \ldots f$, a variance bound three times larger than (4) is obtained [5].

Unfortunately, in high-speed data links, high frequency components due to echo and crosstalk signals can significantly tighten the constraint on timing skew. We therefore estimate the skew requirements for our design using the conservative result of (4), plotted in Fig. 5 for an interleaving factor of $N = 8$. As evident from this illustration, sub-picosecond skew

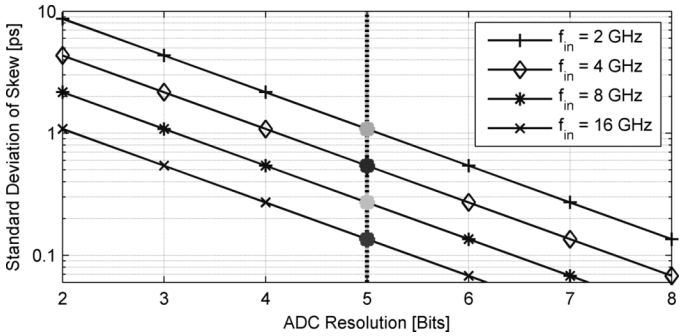

Fig. 5. Bounds on ADC resolution for different input frequencies ($N = 8$).

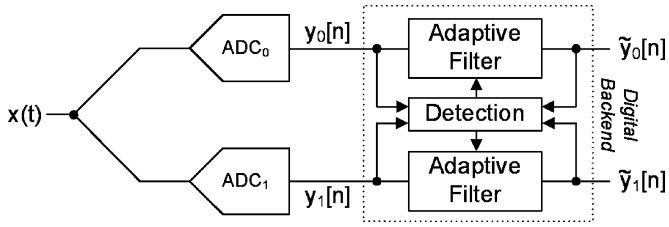

Fig. 6. Digital timing skew correction.

is required with input frequencies larger than 4 GHz, for an ADC resolution of 5 bits. This bound tightens significantly with higher resolution ADCs, and can easily become less than 100 fs for state-of-the-art designs.

In a realistic chip design, both transistor variations and load and trace variations affect the signal and clock paths, and timing skews on the order of several picoseconds [6] are typically unavoidable. In the next sub-section, we therefore discuss the various methods through which designers can mitigate the resulting impairments.

B. Mitigation of Timing Skew

A time-interleaved ADC can be made insensitive to the effect of timing skew by adding a single track-and-hold in front of the sub-ADC array [1], [7]. This front-rank track-and-hold runs at the aggregate clock frequency and creates staircase waveforms at the inputs of the sub-ADCs. Therefore, when each sub-ADC samples the input signal, it samples a nearly constant voltage and can accept some timing skew in its sampling point. However, this solution tends to be only practical in moderate speed designs due to speed limitations with the track-and-hold.

There are two main techniques for compensating the effects of timing skew. The first operates in the digital domain, employing a digital processor at the outputs of the sub-ADCs [8]. Fig. 6 displays an example with two sub-ADCs, in which the outputs $y_0[n]$ and $y_1[n]$ each pass through an adaptive filter that corrects the effects of timing skew. The adaptive filter is a fractional delay filter [9] that interpolates between sub-ADC samples. Typically, such a filter has a large number of filter taps, and the power consumption and complexity of the digital processor is a limiting barrier for this scheme. Although this approach can be attractive for lower frequency designs, multi-GS/s ADCs would suffer a large power penalty that currently makes fractional delay filtering impractical in serial links.

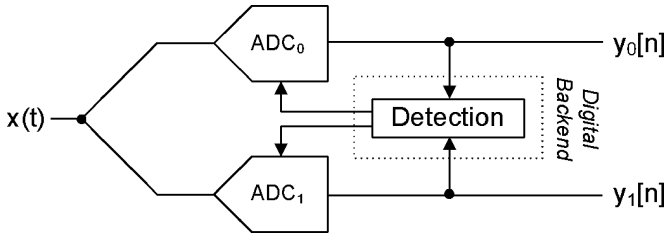

Fig. 7. Mixed-signal timing skew correction.

The second technique for compensating the effects of timing skew follows a mixed-signal approach (see Fig. 7). It uses a digital processor to detect certain characteristics of the discrete-time output and then adjusts analog circuits to eliminate the effects of timing skew [10]. This approach does not suffer from the large at-speed computational burden of a fully digital approach and was therefore chosen for this design.

Both of the above-discussed techniques require calibration, which can be implemented in either the foreground or the background. Foreground calibration has its applications, and may be used when circuit parameters do not vary much with environmental changes, such as voltage or temperature, or when the application allows the ADC to be intermittently taken offline for calibration, such as in oscilloscopes [10]. However, in applications where circuit parameters do vary or where disconnecting the ADC is not an option, such as in serial links, foreground calibration is not a practical solution. Background calibration enables continuous calibration of the correction parameters with the ADC running uninterrupted in normal operation.

Most background calibration techniques for timing skew published to date suffer from various constraints imposed on the spectra and statistical properties of the input signal (see e.g., [11]), which are not always guaranteed in wireline systems. The method presented in the following section greatly relaxes the input signal constraints, and results in a solution that has only a marginal power increase compared to existing solutions.

IV. Background Calibration Algorithm

As derived in [5], the relationship between *SNR* and timing skew is a function of the input signal autocorrelation. This can be rewritten in terms of the values of the timing skew that maximize the SNR, which results in

$$\arg\left(\max \mathrm{SNR}\right) = \arg\left(\sum_{i=0}^{N-1} \max R\left(\tau_i\right)\right). \tag{5}$$

Thus, the SNR is maximized when the autocorrelation of the individual sub-ADCs is maximized. This maximum is achieved with $\tau_i = 0$ for all $i = 0, \ldots, N-1$. The autocorrelation is replaced by the crosscorrelation between each sub-ADC and an additional calibration ADC, which achieves its maximum when both ADCs sample the input signal at the same instant. The calibration ADC is added to the time-interleaved ADC as shown in Fig. 8. It samples the same input signal as the sub-ADCs, and feeds its output to the digital skew detection processor.

The interaction between the calibration ADC and each sub-ADC is further explained by focusing on just one sub-ADC,

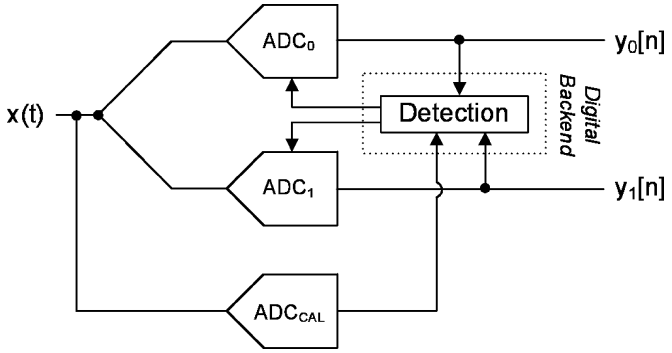

Fig. 8. Adding an auxiliary ADC to a 2-channel time-interleaved ADC.

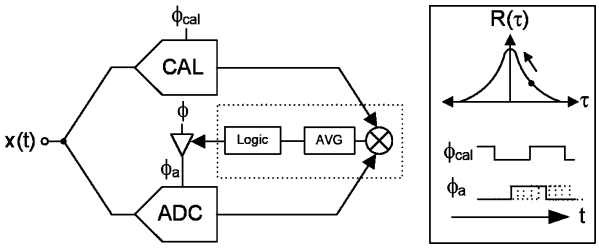

Fig. 9. Timing adjustment to maximize the correlation between a sub-ADC and the auxiliary ADC.

as in Fig. 9. The digital processor estimates a value of the cross-correlation by averaging the product of the sub-ADC and calibration ADC outputs. The value of this crosscorrelation is a function of the timing difference between the sampling edges of the two ADCs, and is maximized when the timing difference is reduced to zero. The main objective of the digital backend is to maximize the value of the crosscorrelation by climbing up the correlation curve. This is achieved with the addition of a delay line such that the sampling edge of the sub-ADC is adjusted in a direction that minimizes the timing difference and maximizes the correlation, as indicated in Fig. 9. LMS-based algorithms can be used to adjust the control words of the delay line, as explained in [12].

Since the crosscorrelation between the sub-ADC and the calibration ADC does not require the transfer function of both ADCs to be identical, it is possible to reduce the resolution of the calibration ADC to a single bit. In the implementation presented in this paper, the calibration ADC consists of a single comparator.

Furthermore, the calibration ADC does not require a sample rate equal to that of the sub-ADC, as long as its sampling edges coincide with the ideal sampling points of the sub-ADC. This allows the above-described scheme to extend to all N interleaved sub-ADCs, such that the architecture includes one additional calibration comparator in total and N additional delay lines, one for each sub-ADC, as in Fig. 1. In order to cycle through and calculate the crosscorrelation for each sub-ADC, the calibration comparator is clocked such that its sampling edges form a timing grid that match the ideal sampling points of each of the sub-ADCs, thus serving as a timing reference. This is achieved by choosing a calibration frequency of f_s/M, where f_s is the time-interleaved ADC sample rate and M is such that the greatest common divisor between N and M is one.

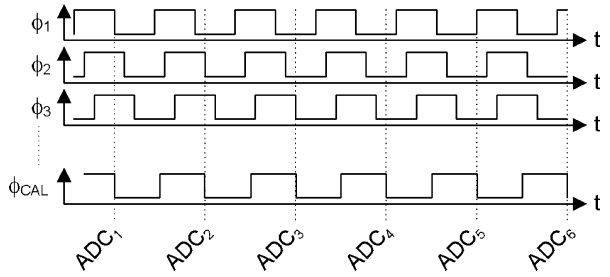

Fig. 10. Sample timing diagram for calibration clock and sub-ADC clocks, assuming an interleaving factor of 8.

Fig. 10 shows an example of such a timing grid, illustrating how the calibration comparator cycles through all the sub-ADCs. With an interleaving factor of eight, the calibration clock has a frequency of $f_s/9$, such that the first sampling edge coincides with the ideal sampling edge of the first sub-ADC, the second sampling edge coincides with the ideal sampling edge of the second sub-ADC, and so on for all eight sub-ADCs. This allows the digital backend to calculate the correlation between each sub-ADC and the calibration comparator. Since each sub-ADC has its own delay line, as shown in Fig. 1, the digital backend maximizes the correlation between each sub-ADC and the calibration comparator by independently adjusting the eight delay lines.

A condition for this algorithm to work is that the correlation must change with the delay line codes; otherwise, the algorithm will not detect a global maximum and thus will not converge. The ADC quantization can limit the types of applicable signals; for example, a sinusoidal input signal with a frequency of f_s would prevent the timing skew from being properly compensated for. However, with sufficiently stochastic signals, as is the case in serial links, quantization is acceptable, as is seen from the extreme example where the sub-ADC also has a single bit resolution [13]. This allows the calibration algorithm to operate with a large set of input signals [12].

A. Clocking the Calibration Comparator

In the implementation of the prototype ADC, which is further described in Section V, an off-chip signal generator was used to create the calibration clock with the required frequency. However, in an SoC environment, this calibration clock ϕ_{cal} can be generated on-chip. The method used to generate the clock depends on the relationship between the frequency $f_{\rm ref}$ of the reference clock $\phi_{\rm ref}$ and the sampling frequency f_s of the time-interleaved ADC.

For example, if $f_{\rm ref} = f_s$, as is possible in lower frequency designs [14], then clock-gating can be used. Fig. 11 shows an example circuit that divides the reference frequency $f_{\rm ref}$ by a factor of three, such that the frequency of the calibration clock is $f_s/3$. When the reference frequency is a fraction of the sampling frequency, such that $f_{\rm ref} = f_s/M$, then an integer or fractional PLL may be used to set the frequency of the calibration clock as desired.

In either realization, the design of the calibration clock generator is simplified because of the algorithm's insensitivity

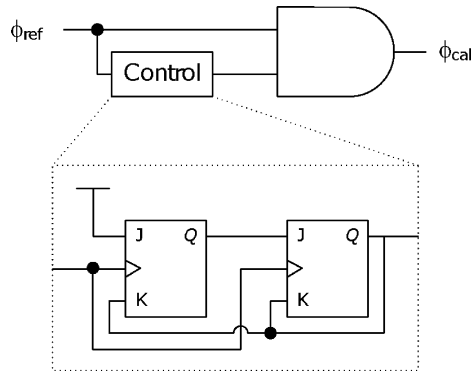

Fig. 11. Using clock-gating to create the calibration clock.

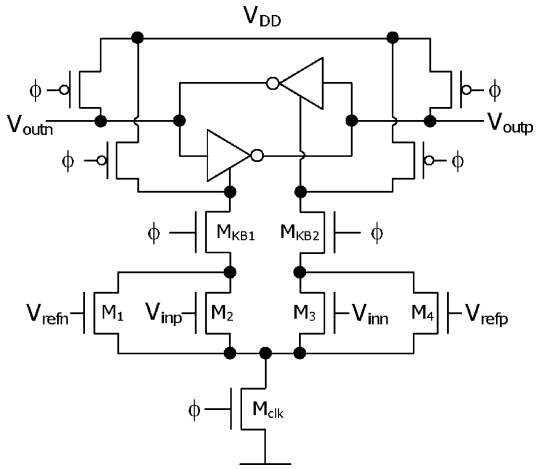

Fig. 12. Schematic of the dynamic comparator.

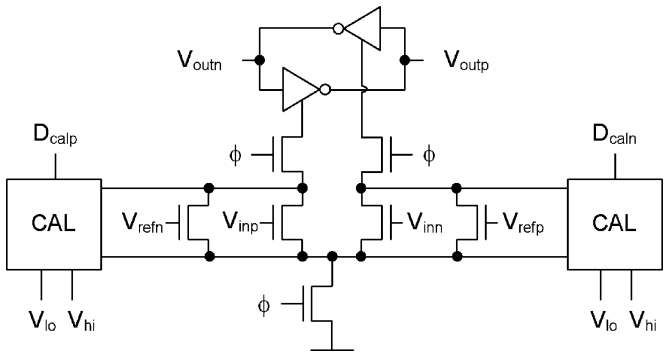

Fig. 13. Comparator with offset correction DAC.

to random calibration clock jitter. The jitter is largely suppressed by the averaging that occurs in calculating the signal correlations.

V. Circuit Design

A prototype ADC was fabricated in a 65 nm process to evaluate the proposed background calibration algorithm. This section discusses the sub-ADC and delay line circuits in detail, and briefly covers the phase generator used in this implementation.

A. The Sub-ADC

The interleaving factor of the time-interleaved ADC affects the sample rate of each individual sub-ADC, and is an important design parameter. Including the effects of input-referred offset of the comparators in the flash ADC, it is possible to obtain a power-optimal interleaving factor, given a certain metastability rate [15]. Since a shallow minimum results [12], an interleaving factor of eight was chosen, which results in a sub-ADC sample rate of 1.5 GS/s, given the target sample rate of 12 GS/s for the complete time-interleaved ADC.

In this design, each sub-ADC consists of a 5-bit flash ADC containing 31 dynamic comparators. Due to the inherent sampling nature of dynamic comparators, it is possible to omit an explicit track-and-hold [16]. However, in this design, we preferred including a passive bootstrapped [17] track-and-hold in each channel to avoid performance degradation due to skew between the comparator clock edges. The reference voltages of the comparators are set through a differential resistor ladder, and their outputs are converted into a 5 bit word using a Wallace Encoder [18].

1) Comparator Design: Fig. 12 shows the schematic of the dynamic comparator, which was used due to its superior input sensitivity, higher energy efficiency, and smaller latching time-constant when compared to a CML latch [19], [20]. The two clocked transistors labeled $\mathrm{M_{KB1}}$ and $\mathrm{M_{KB2}}$ are added to reduce kickback. Kickback noise affects the reference levels provided by the resistor ladder. A power-inefficient approach to reduce kickback is to decrease the resistor ladder impedance, which decreases the time constant on the reference nodes and allows the reference voltages to settle in time for the next cycle. Placing $\mathrm{M_{KB1}}$ and $\mathrm{M_{KB2}}$ between the cross-coupled inverters and the input transistors prevents the precharge of the drain nodes of the input and reference transistors, and thus reduces the swing on the reference nodes [21].

2) Offset Calibration: Reducing the power consumption of the sub-ADCs is achieved by using smaller transistors in the comparators, as that reduces the node capacitances and the dynamic power. A side benefit of this is the reduced input capacitance. However, increased variations, such as those in the transistor threshold voltage [22], result in an increase in input-referred offset, which will degrade the flash ADC performance. In this design, this issue is managed using offset correction [23]. As shown in Fig. 13, a calibration DAC is placed in parallel to the input and reference transistors and injects current through the two comparator branches to compensate for offset. A 5-bit DAC with three binary encoded bits and two thermometer encoded bits was used (see Fig. 14), and was designed for approximately $\mathrm{LSB}/4$ of residual offset, where each $\mathrm{LSB} = 19\ \mathrm{mV}$. Given that there are 8×31 comparators in this design, a total of 248 calibration DACs are used.

Due to the parallel placement of the calibration DAC, the relatively large LSB size, and the ratiometric operation of the dynamic comparator, the digital code required by the calibration DAC to compensate for offset is, to first-order, temperature independent. Thus, a foreground offset calibration approach was adopted. Fig. 15 shows a high-level view of the offset correction algorithm. During calibration, the input and reference nodes are

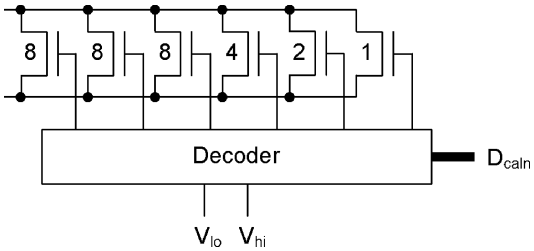

Fig. 14. Offset correction DAC.

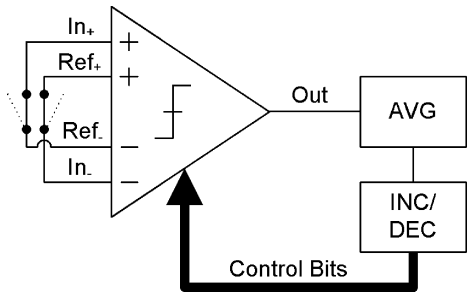

Fig. 15. Foreground offset correction.

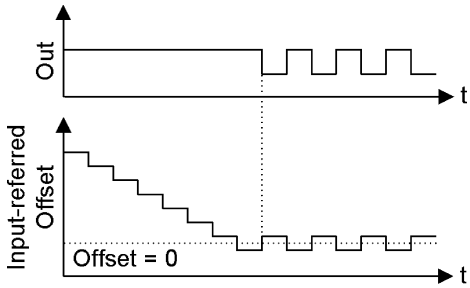

Fig. 16. Timing diagram for foreground offset correction.

shorted such that, in the absence of offset, the comparator is biased at its switching point.

Based on the average of the comparator outputs, the correction DAC code is either incremented or decremented in a direction that reduces the input-referred offset, as in the timing diagram of Fig. 16. Once the input-referred offset changes sign, the comparator output begins to oscillate and the offset is minimized. The residual offset is mostly determined by the calibration DAC LSB.

3) Wallace Encoder: The outputs of the 31 comparators are converted into a 5-bit word with an encoder. A Wallace Encoder [18], which is a ones-adder that sums the outputs of all the comparators, has improved performance over other common encoders [24]. The power of the Wallace Encoder increases exponentially in ADC resolution due to the increase in the required number of full-adders. However, for 5-bit resolution in a 65 nm technology, the simulated power dissipation is less than 1 mW, making it a viable block in a low power ADC. A $(2^N-1)-N$ Wallace Encoder is recursively built with $(2^{N-1}-1)-(N-1)$ Wallace Encoders, the basic unit of which is a full-adder.

B. Delay Line

The delay line closes the feedback loop in the skew calibration algorithm for each sub-ADC, as discussed in Section IV.

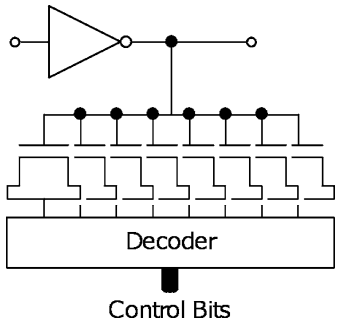

Fig. 17. Delay cell with variable capacitive load.

Since the delay line is placed directly in the clock path, its design is crucial as ADC dynamic performance is a function of the sampling edge jitter.

Each delay line consists of a series of cascaded delay cells, which has a CMOS inverter as its basic building block. The CMOS inverter is used because of its lower power consumption when compared to CML delay cells. However, an inverter-based delay cell suffers from thermal and power-supply noise jitter. Although thermal jitter can be reduced through proper design, power supply variations are problematic, and can result in several picoseconds of jitter. A common solution to mitigate this is to regulate the voltage supply. In this implementation, separate supply and ground lines were provided from off-chip. Slower variations in the delay cell properties, such that those due to temperature, are dealt with via the background timing skew calibration algorithm.

The delay of the inverter is adjusted by loading it with a variable capacitor. To first-order, the change in the inverter delay, assuming a switching point at half the voltage supply, is

$$\Delta t_d = \Delta C_L \frac{V_{\rm DD}}{2 \cdot I_{inv}} \tag{6}$$

where ΔC_L is the change in the load capacitance and I_{inv} is the inverter drive strength. The value of the capacitive change and the inverter drive strength are co-designed to meet monotonicity specifications as well as to minimize power.

1) Capacitive Load: The variable capacitive load at the output of each delay cell consists of MOS transistors [10], since the gate capacitance of MOS transistors is a function of bias voltages. Shorting the drain and source nodes of the transistor and digitally controlling this shorted node changes the value of the gate capacitance, and more importantly the delay of an inverter that is loaded with the gate of the transistor. Creating an array of such digitally controlled transistors results in a digitally controlled capacitive load. A 7-bit segmented load with 5 binary encoded bits and 2 thermometer encoded bits was used (see Fig. 17).

2) Cascaded Delay Cells: In order to compensate for the expected timing skew from the clocking network and the sub-ADCs and to limit the change in delay per delay cell to a fraction of its delay, a series of three cascaded delay cells is used in the delay line. Each delay cell is controlled by the same control word, as shown in Fig. 18, to improve monotonicity. The total simulated correction range was 32 ps with a step size of approximately 0.25 ps.

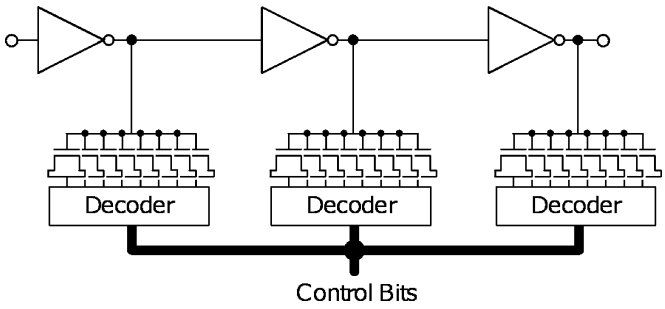

Fig. 18. Complete delay line.

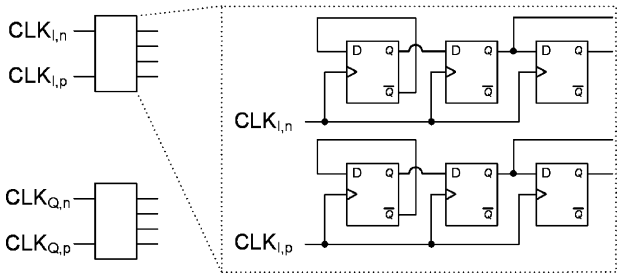

Fig. 19. Phase generator for sub-ADC clocks.

C. Phase Generator

The phases for the eight sub-ADCs are ideally spaced with phase offsets of 45° such that the input signal is uniformly sampled. Although a common technique is to use a PLL or DLL, this implementation uses shift registers [25]. The input to the phase generator consists of two 3 GHz clocks that have a phase offset of 90°, and each differential clock passes through a series of shift registers as in Fig. 19. The resulting outputs consist of eight 1.5 GHz clocks with the required phase offsets. The outputs of the phase generator are then followed by the digitally controlled delay lines described above.

VI. Measurement Results

The prototype ADC was fabricated in the TSMC 65 nm GP process and packaged in a QFN-48 package. The die photo is shown in Fig. 20, with the eight sub-ADCs outlined in black. The chip has a total area of 1.3 mm^2 and an active area of 0.44 mm^2.

A. Static Performance

The differential (DNL) and integral (INL) nonlinearity were measured by collecting the histogram of a low frequency input signal [26] for a single sub-ADC, since time-interleaving would otherwise average the DNL and INL performance metrics [27]. The comparator offsets in the flash sub-ADCs are a limiting factor in the static performance, which is why foreground offset calibration was used (see Section V). The typical DNL and INL of a single sub-ADC before foreground offset correction are shown in Figs. 21 and 22. The typical DNL and INL after foreground offset correction are shown in Figs. 23 and 24. Both the DNL and INL have been reduced to less than ± 0.5 LSB, which validates the functionality of the foreground offset correction.

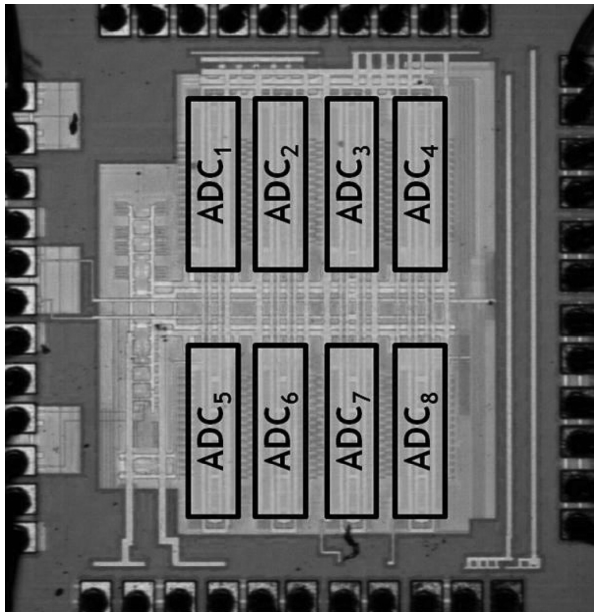

Fig. 20. Die photo.

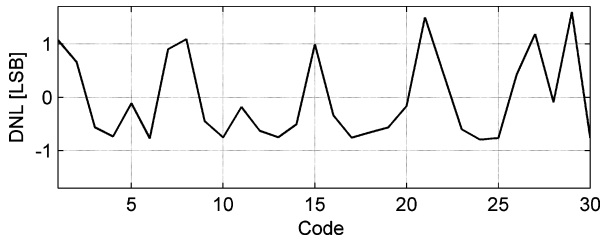

Fig. 21. DNL before offset calibration.

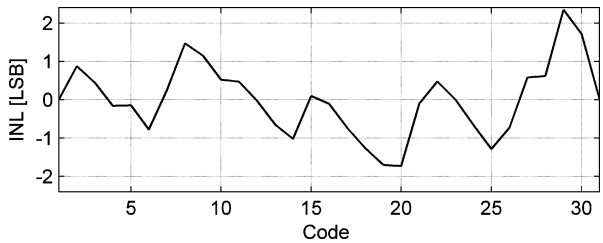

Fig. 22. INL before offset calibration.

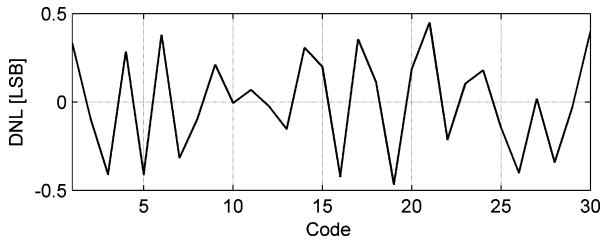

Fig. 23. DNL after offset calibration.

B. Timing Skew Calibration

The background timing skew calibration algorithm is run off-chip using Matlab [12]. The ADC outputs are decimated by a factor of 81 and are read via a data capture card. Control codes are sent back to the delay cells in the prototype ADC. These

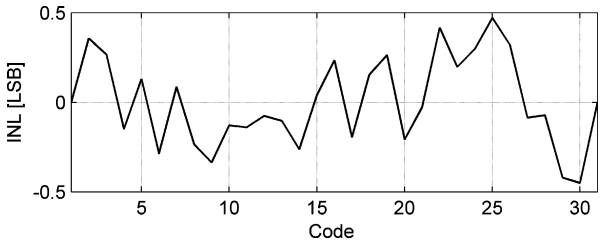

Fig. 24. INL after offset calibration.

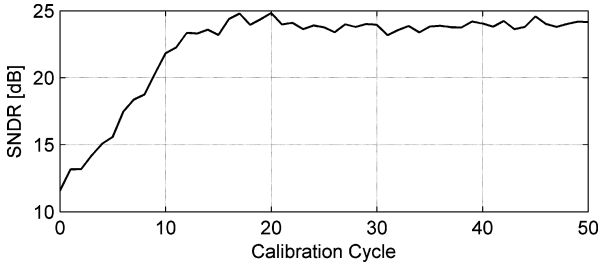

Fig. 25. SNDR during convergence of the timing skew calibration algorithm with 500,000 samples per calibration cycle.

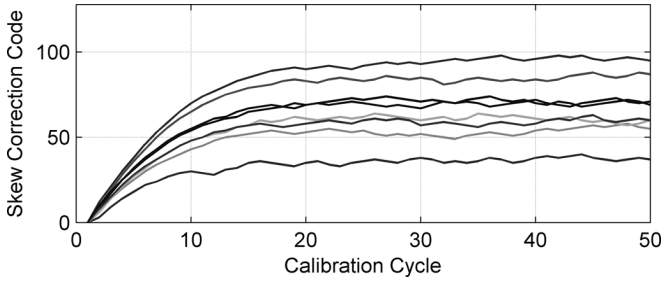

Fig. 26. Timing skew correction code convergence corresponding to the measurement of Fig. 25.

codes are updated once every calibration cycle, which consists of K samples. The time in seconds the calibration cycle spans is a function of K and the frequency of the calibration clock. The required number of calibration cycles, and thus the total time required to converge, is also a function of the calibration algorithm used.

In the following results, a calibration clock frequency of 480 MHz, an input frequency of 8 GHz, and a sampling frequency of 12 GS/s were used. An off-chip balun (Picosecond model 5315A) was used with a high-speed generator to create the input and clock signals. In Fig. 25, the measured SNDR improves from 12 dB to approximately 24 dB, once the calibration is turned on, and converges within 20 calibration cycles. Fig. 26 shows the timing skew delay codes used by each of the eight delay lines as a function of the calibration cycle. Since 500,000 samples were used, a total time of approximately 160 ms is required for convergence. This is reduced by using fewer samples, which would also result in a noisier convergence value due to the uncertainty in the approximation of the correlation, as in Fig. 27. In this example, only 50,000 samples per calibration cycle were used, and thus requires 16 ms to converge. However, the SNDR varies by over 3 dB across calibration cycles.

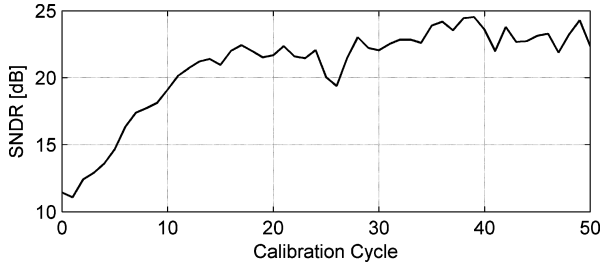

Fig. 27. SNDR during convergence of the timing skew calibration algorithm with 50,000 samples per calibration cycle.

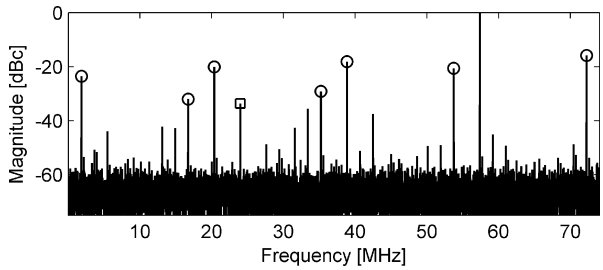

Fig. 28. Decimated output spectrum without timing skew calibration.

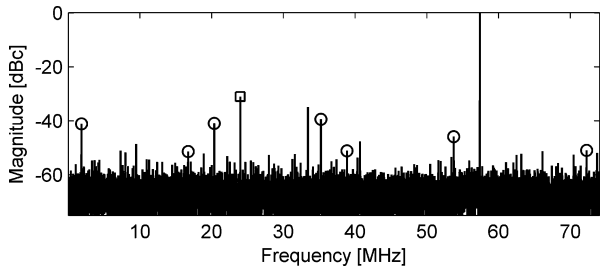

Fig. 29. Decimated output spectrum with timing skew calibration.

C. Dynamic Performance

1) Output Spectrum: The output spectrum, with a decimation factor of 81, of the time-interleaved ADC with an 8 GHz input signal is shown in Figs. 28 and 29, with and without timing skew calibration. A frequency larger than $f_s/2$ is used to demonstrate the relaxed bandwidth constraints on the input signal. Without timing skew calibration, the decimated output spectrum is limited by the spurs due to timing skew, as denoted by the circles in Fig. 28. The third harmonic is marked with a square and has a magnitude less than that of the timing skew spurs. Fig. 29 shows the decimated output spectrum with timing skew calibration. The spurs due to timing skew have dropped 10–30 dB, and the SFDR is limited by the third harmonic at a magnitude of -31 dBc.

2) Input Frequency Sweep: The measured SNDR as a function of the input frequency is plotted in Fig. 30 with and without timing skew calibration. The curves have identical values at low frequencies due to the negligible sampling error caused by timing skew. The SNDR of the ADC without calibration drops approximately 15 dB as the input frequency increases to 8 GHz. This curve flattens out with calibration turned on, such that the SNDR of the ADC only suffers a 3 dB performance drop between low and high frequency inputs. The SNDR at Nyquist is 25.1 dB and the performance improvement at 8 GHz is 12 dB.

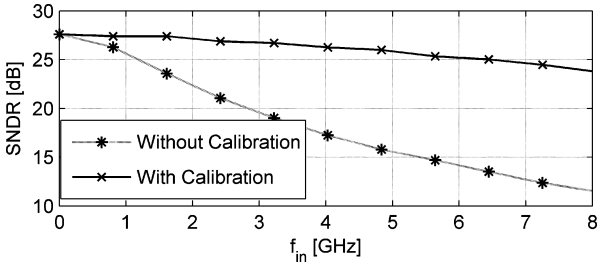

Fig. 30. Input frequency sweep and SNDR performance with and without calibration.

TABLE I
PERFORMANCE SUMMARY OF PROTOTYPE ADC

Parameter	Value	
Process	TSMC 65 nm GP	
Active Area	0.44 mm^2	
VDD	1.1 V	
Full Scale Range	590 mV	
Resolution	5 b	
Sample Rate	12 GS/s	
Input Capacitance	1.1 pF	
	f_{in} = 10 MHz	f_{in} = 6 GHz
SNDR	27.5 dB	25.1 dB
FOM	0.35 pJ/conv-step	0.46 pJ/conv-step
Power	81 mW (excluding digital backend, I/O cells, and input clock buffers)	

The residual timing skew after calibration is calculated to be approximately 0.4 ps and the timing jitter is estimated to be 0.6 psrms.

Table I summarizes the performance of the ADC. The power consumption of the prototype ADC is 81 mW, excluding the digital backend, the I/O cells, and the input clock buffer. The SNDR at low and high frequencies is 27.5 dB and 25.1 dB, respectively. This corresponds to a figure of merit [28], as calculated with

$$\text{FOM} = \frac{P}{f_s \cdot 2^{\text{ENOB}}} \tag{7}$$

of 0.35 pJ/conv-step and 0.46 pJ/conv-step for low and high input frequencies, respectively.

D. Comparisons

Fig. 31 plots the conversion energy (P/f_s,) versus SNDR for ADCs published at the IEEE International Solid-State Circuits Conference (ISSCC) and the VLSI Circuits Symposium with a sample rate of over 10 GS/s [29]. In this data set, this work is the most power-efficient ADC with a sample rate larger than 10 GS/s published to date. All of the data points plotted in Fig. 31 are listed in Table II.

VII. CONCLUSION

This work demonstrates the design of a low-power multi-GS/s time-interleaved ADC applicable to serial links applications with tight power bounds. A background calibration algorithm that mitigates the effect of timing skew is introduced and validated. The power consumption of the ADC is reduced with the use of hundreds of trim circuits that individually correct the offset of each of the comparators, allowing smaller transistors be used. The 12 GS/s 5-bit proof-of-concept ADC has the best power-efficiency when compared to other published works with sample rates larger than 10 GS/s.

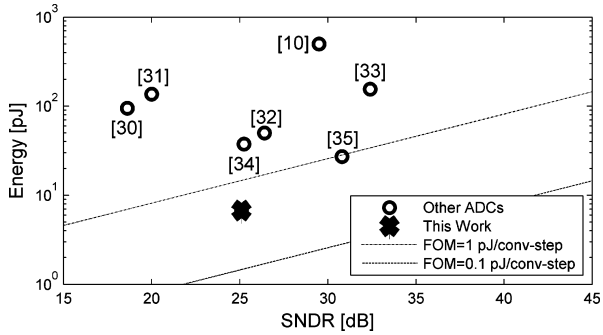

Fig. 31. Comparison with ADCs sampling faster than 10 GS/s.

TABLE II
PUBLISHED ADCS (ISSCC AND VLSI) FASTER THAN 10 GS/S

Reference	Resolution [Bits]	Sample Rate [GS/s]	Power [W]	SNDR [dB]	Technology
[10]	8	20	9	29.5	0.18 um
[30]	3	40	3.8	18.6	HBT
[31]	5	22	3	20	SiGe
[32]	6	24	1.2	26.4	90 nm
[33]	6	10.3	1.6	32.4	90 nm
[34]	6	40	1.5	25.2	65 nm
[35]	6	16	0.435	30.8	65 nm
This Work	5	12	0.081	25.1	65 nm

ACKNOWLEDGMENT

The authors thank K. Ring of Intersil for help with place/route and timing verification, and R. Payne of Texas Instruments for help and guidance in testing.

REFERENCES

[1] S. Gupta, M. Inerfield, and J. Wang, "A 1-GS/s 11-bit ADC with 55-dB SNDR, 250-mW power realized by a high bandwidth scalable time-interleaved architecture," *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2650–2657, Dec. 2006.

[2] M. Harwood *et al.*, "A 12.5 Gb/s SerDes in 65 nm CMOS using a baud-rate ADC with digital receiver equalization and clock recovery," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2007, vol. 1, pp. 436–591.

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

[4] P. Huang and Y. Chiu, "A gradient-based algorithm for sampling clock skew calibration of SHA-less pipeline ADCs," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, 2007, pp. 1241–1244.

[5] M. El-Chammas and B. Murmann, "General analysis on the impact of phase-skew in time-interleaved ADCs," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 56, no. 5, pp. 902–910, May 2009.

[6] A. Agrawal *et al.*, "An 8 × 5 Gb/s parallel receiver with collaborative timing recovery," *IEEE J. Solid-State Circuits*, vol. 44, no. 11, pp. 3120–3130, Nov. 2009.

[7] K. Poulton, J. Corcoran, and T. Hornak, "A 1-GHz 6-bit ADC system," *IEEE J. Solid-State Circuits*, vol. SC-22, no. 6, pp. 962–970, Dec. 1987.

[8] S. Jamal *et al.*, "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.

[9] T. Laakso *et al.*, "Splitting the unit delay – tools for fractional delay filter design," *IEEE Signal Process. Mag.*, vol. 13, no. 1, pp. 30–60, Jan. 1996.
[10] K. Poulton *et al.*, "A 20 GS/s 8b ADC with a 1 MB memory in 0.18 μm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2003, vol. 1, pp. 318–496.
[11] D. Camarero *et al.*, "Mixed-signal clock-skew calibration technique for time-interleaved ADCs," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 55, no. 12, pp. 3676–3687, Dec. 2008.
[12] M. El-Chammas, "Background calibration of timing skew in time-interleaved A/D converters," Ph.D. dissertation, Stanford Univ., Stanford, CA, 2010.
[13] J. McFadden, "The axis-crossing intervals of random functions," *IRE Trans. Inf. Theory*, vol. 2, no. 4, pp. 146–150, Dec. 1956.
[14] S. Louwsma *et al.*, "A 1.35 GS/s, 10 b, 175 mW time-interleaved AD converter in 0.13 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 778–786, Apr. 2008.
[15] H. Veendrick, "The behaviour of flip-flops used as synchronizers and prediction of their failure rate," *IEEE J. Solid-State Circuits*, vol. SC-15, no. 2, pp. 169–176, Apr. 1980.
[16] G. Van der Plas, S. Decoutere, and S. Donnay, "A 0.16 pJ/conversion-step 2.5 mW 1.25 GS/s 4 b ADC in a 90 nm digital CMOS process," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2006, vol. 1, p. 2310.
[17] A. Abo and P. 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.
[18] C. S. Wallace, "A suggestion for a fast multiplier," *IEEE Trans. Electronic Computers*, vol. EC-13, no. 1, pp. 14–17, Feb. 1964.
[19] T. Toifl, "Design techniques for ultra-low-power and compact transceivers in CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC)*, 2008, ATAC Design Forum: Future of High-Speed Transceivers.
[20] T. Toifl *et al.*, "A 22-Gb/s PAM-4 receiver in 90-nm CMOS SOI technology," *IEEE J. Solid-State Circuits*, vol. 41, no. 4, pp. 954–965, Apr. 2006.
[21] T. Sundstrom and A. Alvandpour, "A kick-back reduced comparator for a 4–6-bit 3-GS/s flash ADC in a 90 nm CMOS process," in *Proc. Int. Conf. Mixed Design of Integrated Circuits and Systems*, 2007, pp. 195–198.
[22] M. Pelgrom, A. Duinmaijer, and A. Welbers, "Matching properties of MOS transistors," *IEEE J. Solid-State Circuits*, vol. 24, no. 5, pp. 1433–1439, Oct. 1989.
[23] K.-L. Wong and C.-K. Yang, "Offset compensation in comparators with minimum input-referred supply noise," *IEEE J. Solid-State Circuits*, vol. 39, no. 5, pp. 837–840, May 2004.
[24] E. Sail and M. Vesterbacka, "Thermometer-to-binary decoders for flash analog-to-digital converters," in *Proc. Eur. Conf. Circuit Theory and Design*, 2007, pp. 240–243.
[25] X. Gao, B. Nauta, and E. Klumperink, "Advantages of shift registers over DLLs for flexible low jitter multiphase clock generation," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 55, no. 3, pp. 244–248, Mar. 2008.
[26] M. Bossche, J. Schoukens, and J. Renneboog, "Dynamic testing and diagnostics of A/D converters," *IEEE Trans. Circuits Syst.*, vol. 33, no. 8, pp. 775–785, Aug. 1986.
[27] J. Simoes, J. Landeck, and C. Correia, "Nonlinearity of a data-acquisition system with interleaving/multiplexing," *IEEE Trans. Instrum. Measure.*, vol. 46, pp. 1274–1279, Dec. 1997.
[28] R. Walden, "Performance trends for analog to digital converters," *IEEE Commun. Mag.*, vol. 37, no. 2, pp. 96–101, Feb. 1999.
[29] B. Murmann, ADC Performance Survey 1997–2010. Stanford Univ. [Online]. Available: http://www.stanford.edu/_murmann/adcsurvey.html
[30] W. Cheng *et al.*, "A 3 b 40 GS/s ADC-DAC in 0.12 μm SiGe," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2004, vol. 1, pp. 262–263.
[31] P. Schvan *et al.*, "A 22 GS/s 5 b ADC in 0.13 μm SiGe BiCMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2006, vol. 1, pp. 2340–2349.
[32] P. Schvan *et al.*, "A 24 GS/s 6 b ADC in 90 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2008, vol. 1, pp. 544–634.
[33] A. Nazemi *et al.*, "A 10.3 GS/s 6 bit (5.1 ENOB at Nyquist) time-interleaved pipelined ADC using open-loop amplifiers and digital calibration in 90 nm CMOS," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2008, pp. 18–19.
[34] Y. Greshishchev *et al.*, "A 40 GS/s 6 b ADC in 65 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2010, vol. 1, pp. 390–391.
[35] C.-C. Huang, C.-Y. Wang, and J.-T. Wu, "A CMOS 6-Bit 16-GS/s time-interleaved ADC with digital background calibration," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2010, pp. 159–160.

Manar El-Chammas (S'04–M'10) received the B.E. degree in computer and communications engineering from the American University of Beirut, Lebanon, in 2004, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 2006 and 2010, respectively.

He has held internship positions with the Advanced Development Lab, LSI Corporation, the IBM Zurich Research Lab, and Kilby Labs in Texas Instruments. He recently joined Texas Instruments, Inc., Dallas, TX, and works on high-speed data converters.

Boris Murmann (S'99–M'03–SM'09) received the Dipl.-Ing. (FH) degree in communications engineering from Fachhochschule Dieburg, Dieburg, Germany, in 1994, the M.S. degree in electrical engineering from Santa Clara University, Santa Clara, CA, in 1999, and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 2003.

From 1994 to 1997, he was with Neutron Mikrolektronik GmbH, Hanau, Germany, where he developed low-power and smart-power ASICs in automotive CMOS technology. Since 2004 he has been with the Department of Electrical Engineering, Stanford University, Stanford, CA, where he currently serves as an associate professor. His research interests are in the area of mixed-signal integrated-circuit design, with special emphasis on data converters and sensor interfaces.

In 2008, Dr. Murmann was a co-recipient of the Best Student Paper Award at the VLSI Circuits Symposium and the recipient of the Best Invited Paper Award at the IEEE Custom Integrated Circuits Conference (CICC). In 2009, he received the Agilent Early Career Professor Award. He currently serves as an associate editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS and as a member of the IEEE International Solid-State Circuits Conference (ISSCC) program committee.