Jitter Measurement and Compensation for Analog-to-Digital Converters

Chi-Wei Fan, *Student Member, IEEE*, and Jieh-Tsorng Wu, *Senior Member, IEEE*

***Abstract*—Clock jitter is measured and digitized by a stochastic time-to-digital converter (TDC). This jitter information is used to compensate the sampling error of an analog-to-digital converter (ADC) caused by the clock jitter. The following two system scenarios are covered: 1) an ADC with a clean external clock and 2) an ADC with an external clock as the main jitter source. TDC calibrations for both scenarios are proposed. The calibrations are based on signal reconstruction and can be performed in the background. Both theoretical analyses and system simulations are provided to verify the proposed jitter compensation and TDC calibration techniques.**

***Index Terms*—Analog-to-digital conversion, calibration, clocks, jitter, sampling methods, signal reconstruction, signal sampling, time measurement.**

I. Introduction

A Nyquist-rate analog-to-digital converter (ADC) first samples a continuous-time analog signal and then quantizes the sampled data into a discrete-time digital sequence. A periodic clock is required to provide a reference for the sampling time. If the sampling clock jitters, a sampling error occurs during the sampling process [1], [2]. Excess clock jitter can degrade the signal-to-noise ratio (SNR) performance of an ADC. For an ADC with more than 100 MS/s sampling rate, if the analog input is a 50-MHz sine wave and the clock jitter is random, the root-mean-square (rms) value of the clock jitter must be less than 0.32 ps to ensure an SNR greater than 80 dB. Clocks generated from phase-locked loops can hardly achieve this stringent jitter requirement [3]. Low-jitter clocks, which are mostly based on crystal oscillators, are inflexible and expensive.

It is possible to relax the clock jitter requirement by introducing jitter compensation in the analog-to-digital signal path. The scheme of Tourabaly and Osseiran [4] modulates the analog input before the sampler so that the correct input signal is sampled. However, this scheme requires high-precision analog circuits, which are difficult to implement. The use of differentiators also makes it sensitive to high-frequency noises.

In this paper, we propose a technique that employs digital signal processing to relax the clock jitter requirement. Clock jitter is measured and digitized by a stochastic time-to-digital converter (TDC) [5]. This jitter information is then used to compensate the ADC's sampling error in the digital domain, improving the ADC's SNR performance. We also propose techniques for TDC calibration. The calibration can be performed in the background without interrupting the normal ADC operation. Both theoretical analyses and system simulations are provided to verify the proposed jitter compensation and TDC calibration techniques. A 16-bit 100-MS/s ADC system is discussed and simulated as a design example.

The rest of this paper is organized as follows. Section II introduces the principle of jitter compensation. Section III describes the concept of signal reconstruction. The proposed TDC calibrations are based on signal reconstruction. Section IV gives an overview of the stochastic TDC. Section V describes a jitter compensation scenario in which the external clock is clean. Section VI describes another jitter compensation scenario in which the external clock is the main jitter source. The two scenarios employ different TDC calibration schemes. Section VII discusses some hardware implementation issues of the proposed techniques. Finally, Section VIII draws conclusions.

II. Jitter Compensation

The basic principle of the proposed jitter compensation is illustrated in Fig. 1. An ADC samples and quantizes the analog signal $V_i(t)$ and generates the corresponding digital sequence $D_i[k]$. A clock dictates the instants at which $V_i(t)$ is sampled. The kth sampling time is $(k+\epsilon[k])T_s$, where T_s is the nominal sampling interval, and $\epsilon[k]$ is the clock jitter normalized to T_s. The clock jitter at the kth sample is $\Delta t[k]=\epsilon[k]T_s$. A jitter-to-digital converter (JDC) measures the $\epsilon[k]$ jitter and produces a jitter estimation $\hat{\epsilon}[k]$ in digital form. The relationship between $\epsilon[k]$ and $\hat{\epsilon}[k]$ is defined as

$$\hat{\epsilon}[k]=\epsilon[k]+\epsilon_e[k] \tag{1}$$

where $\epsilon_e[k]$ is the JDC measurement error. A jitter compensation filter (JCF) uses the $\hat{\epsilon}[k]$ data to correct the sampling error in $D_i[k]$. The corrected output from the JCF is $D_c[k]$.

The ADC's SNR performance is measured by applying the following sine wave input:

$$V_i(t)=A_i\times\sin(\omega_i t+\phi_i) \tag{2}$$

where A_i is the amplitude, ω_i is the input frequency, and ϕ_i is the phase. The corresponding output from the ADC can be expressed as

$$D_i[k]=A_i\sin\left(\Omega_i(k+\epsilon[k])+\phi_i\right)=V_i(kT_s)+q[k]+V_e[k] \tag{3}$$

Manuscript received March 23, 2008; revised July 30, 2008. Current version published October 9, 2009. This work was supported in part by the National Science Council of Taiwan, under Grant NSC-96-2221-E-009-177 and in part by the MediaTek Research Center, National Chiao-Tung University. The Associate Editor coordinating the review process for this paper was Dr. Dario Petri.

The authors are with the Department of Electronics Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, (e-mail: jtwu@mail.nctu.edu.tw).

Digital Object Identifier 10.1109/TIM.2009.2021209

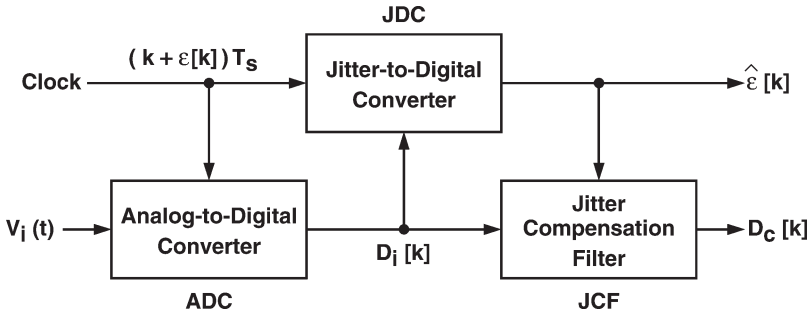

Fig. 1. Jitter compensation block diagram.

where $\Omega_i = \omega_i T_s$ is the normalized input frequency. The $D_i[k]$ signal consists of the following: 1) the desired input $V_i(kT_s)$; 2) the quantization noise $q[k]$; and 3) the sampling error $V_e[k]$ caused by the clock jitter. The SNR of the $D_i[k]$ signal is defined as

$$\text{SNR} \equiv \frac{P_s}{P_q + P_e} \tag{4}$$

where $P_s = (1/2)A_i^2$ is the signal power of $V_i(kT_s)$, P_q is the signal power of the $q[k]$ sequence, and P_e is the signal power of the $V_e[k]$ sequence. Considering an ideal B-bit ADC with an input range of ± 1, it has a uniform quantization step size of $s_q = 2/2^B$. The quantization noise power of an ideal quantization process can be approximated by [6]

$$P_q \equiv \overline{(q[k])^2} = \frac{1}{12} \times s_q^2 = \frac{1}{12} \times \left(\frac{2}{2^B}\right)^2. \tag{5}$$

The quantization noise $q[k]$ is assumed to be random and uniformly spreads between $\pm 1/2^B$. If the clock jitter is small, the sampling error $V_e[k]$ can be approximated by [1]

$$V_e[k] \approx \left.\frac{V_i(t)}{dt}\right|_{t=kT_s} \times \Delta t[k] = A_i \omega_i \cos(\Omega_i k + \phi_i) \times \Delta t[k] \tag{6}$$

where $\Delta t[k] = \epsilon[k] T_s$ is the clock jitter. The sampling error power can be expressed as

$$P_e \equiv \overline{(V_e[k])^2} = \frac{1}{2} A_i^2 \omega_i^2 \times (\Delta t_{\text{rms}})^2 = \frac{1}{2} A_i^2 \Omega_i^2 \times \epsilon_{\text{rms}}^2 \tag{7}$$

where Δt_{rms} is the rms of $\Delta t[k]$, and ϵ_{rms} is the rms of $\epsilon[k]$. The clock jitter is assumed to be random and has a mean of zero. Assume that the input is a full-range sine wave expressed as (2), with $A_i = 1$. From (4), (5), and (7), the SNR becomes

$$\text{SNR}_i = \frac{1}{(2/3)2^{-2B} + \omega_i^2 (\Delta t_{\text{rms}})^2} = \frac{1}{(2/3)2^{-2B} + \Omega_i^2 \epsilon_{\text{rms}}^2}. \tag{8}$$

SNR_i is the maximum SNR performance of the ADC without jitter compensation.

The theory of jitter compensation is discussed as follows. A band-limited signal $V_i(t)$ can be expressed in inverse Fourier transform as

$$V_i(t) = \frac{1}{2\pi} \int_{-\omega_B}^{+\omega_B} V(j\omega) e^{j\omega t} d\omega \tag{9}$$

where $V(j\omega)$ is the Fourier transform of $V_i(t)$, and ω_B is its bandwidth. Neglecting the quantization noise, the $D_i[k]$ signal in Fig. 1 is simply the $V_i(t)$ input sampled at $t = (k + \epsilon[k])T_s$. It can be expressed as

$$D_i[k] = \frac{1}{2\pi} \int_{-\omega_B}^{+\omega_B} V(j\omega) e^{j\omega(kT_s + \epsilon[k]T_s)} d\omega. \tag{10}$$

Assume that only the kth sample $D_i[k]$ contains a sampling error. Then, the sampling error is caused by a jitter $\epsilon[k]$ at the kth sampling. To correct this sampling error, the required JCF is a filter with a frequency-domain transfer function of $e^{-j\omega\epsilon[k]T_s}$. The discrete-time impulse response of this linear-phase filter is

$$\begin{aligned} h_c[n, \epsilon[k]] &= \frac{1}{2\pi} \int_{-\pi}^{+\pi} e^{-j\Omega\epsilon[k]} e^{j\Omega n} d\Omega \\ &= \frac{\sin(\pi(n - \epsilon[k]))}{\pi(n - \epsilon[k])} = \text{sinc}(n - \epsilon[k]) \end{aligned} \tag{11}$$

where $\Omega = \omega T_s$ is the normalized frequency.

Fig. 2 shows a graphic illustration of this jitter compensation scheme. Assume that a sampling error occurs at the kth sampling instant. The $V_i(t)$ is sampled at $t = (k + \epsilon[k])T_s$ instead of $t = kT_s$. The magnitude at point A′ is quantized as $D_i[k]$. Thus, the ADC perceives a different $\hat{V}_i(t)$ input instead of $V_i(t)$. The signal $\hat{V}_i(t)$ has a value of $D_i[k]$ at $t = kT_s$, denoted

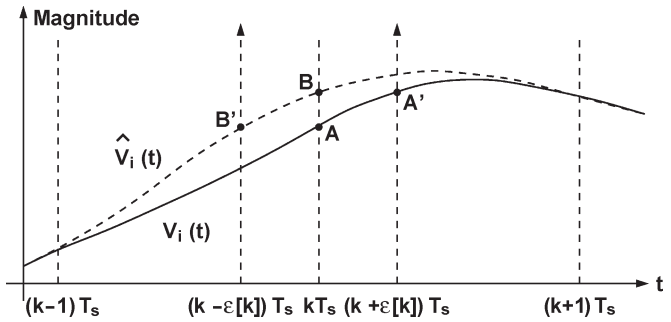

Fig. 2. Jitter compensation principle.

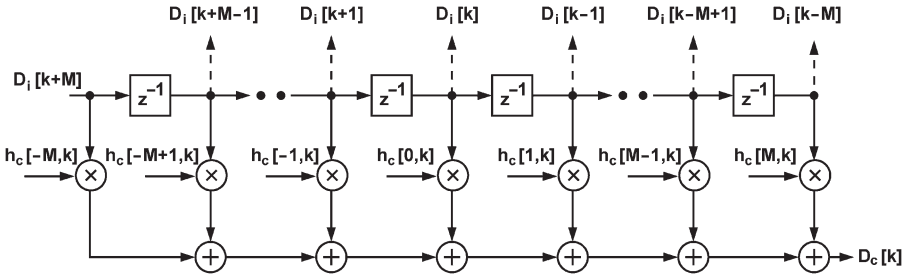

Fig. 3. Jitter compensation filter with $2M + 1$ taps.

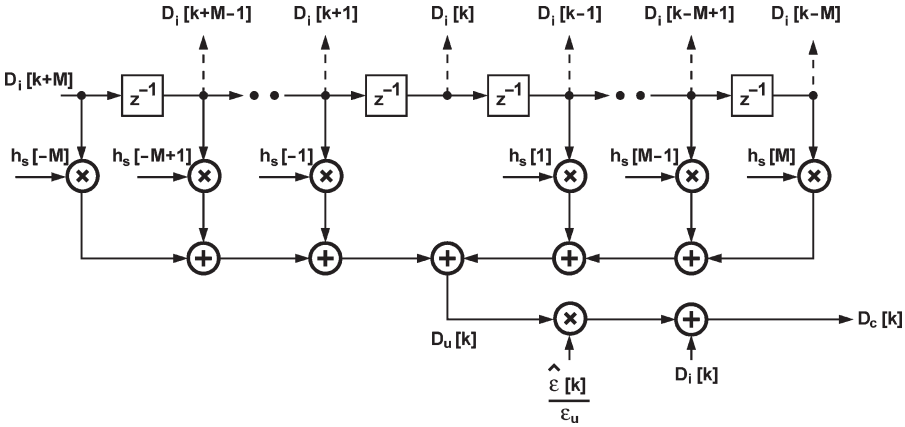

Fig. 4. Simplified jitter compensation filter with $2M + 1$ taps.

as point B. The filter of (11) interpolates the value of $\hat{V}_i(t)$ at $t = (k - \epsilon[k])T_s$, denoted as point B′, which is a correct estimation of $V_i(t)$ at $t = kT_s$, denoted as point A.

Fig. 3 shows a finite-impulse-response (FIR) filter with $2M + 1$ taps that approximates the JCF of (11). Applying the measured $\hat{\epsilon}[k]$ data from the JDC and assuming $\hat{\epsilon}[k] \ll 1$, the filter's output can be expressed as

$$\begin{aligned} D_c[k] &= \sum_{n=k-M}^{k+M} D_i[n] \times h_c\left[k - n, \hat{\epsilon}[k]\right] \\ &\approx D_i[k] + \sum_{n=1}^{M} \left[\frac{1}{2\pi} \int_{-\omega_B}^{+\omega_B} V(j\omega) e^{j\omega k T_s} \right. \\ &\qquad \left. \times \frac{2j \sin(n\omega T_s)}{(-1)^n \times n} \hat{\epsilon}[k] d\omega \right]. \end{aligned} \tag{12}$$

Exchanging the order of integration and summation, (12) becomes

$$D_c[k] \approx D_i[k] + \frac{j\hat{\epsilon}[k]}{\pi} \int_{-\omega_B}^{+\omega_B} V(j\omega) e^{j\omega k T_s} F_c(M, \omega T_s) d\omega \tag{13}$$

where

$$F_c(M, \omega T_s) = \sum_{n=1}^{M} \frac{\sin(n\omega T_s)}{(-1)^n \times n}. \tag{14}$$

Let $\Omega = \omega T_s$, if M approaches infinity, (14) becomes

$$\lim_{M \to \infty} F_c(M, \Omega) = -\frac{\Omega}{2}. \tag{15}$$

Furthermore, if there is no JDC measurement error so that $\hat{\epsilon}[k] = \epsilon[k]$, (13) becomes $D_c[k] = V_i(kT_s)$. This proves that the JCF can compensate the sampling errors and recover the original $V_i(kT_s)$.

In Fig. 3, the tap coefficients $h_c[n, \epsilon[k]]$ are recalculated every clock cycle due to the different $\hat{\epsilon}[k]$s at different ks. The hardware cost of this JCF is very high. It is also difficult for this JCF to achieve high-speed operation. A simplified JCF is proposed as follows. Assuming $\epsilon[k] \ll 1$, (13) can be approximated by

$$D_c[k] \approx D_i[k] + \frac{\hat{\epsilon}[k]}{\epsilon_u} \times D_u[k] \tag{16}$$

where

$$D_u[k] = \sum_{n=k-M}^{k-1} (D_i[n] h_s[k - n]) + \sum_{n=k+1}^{k+M} (D_i[n] h_s[k - n]) \tag{17}$$

with

$$h_s[n] = \mathrm{sinc}(n - \epsilon_u). \tag{18}$$

In the above equations, ϵ_u is a predefined jitter constant. Fig. 4 shows the resulting JCF. In this implementation, once a specific ϵ_u is chosen, the $h_s[n]$ tap coefficients are fixed. The $D_u[k]$ signal of (17) is calculated using a FIR filter with the fixed $h_s[n]$

tap coefficients. The sampling error is estimated by multiplying $D_u[k]$ with the $\hat{\epsilon}[k]/\epsilon_u$ ratio. In Fig. 4, $\hat{\epsilon}[k]$ is only used at one place. The value of ϵ_u can be chosen so that the division by ϵ_u becomes a simple bit-shifting operation in a binary numeral system. Note that the magnitude of $D_u[k]$ can be scaled by varying ϵ_u.

As an example, consider a sine wave input $V_i(t)$ expressed as (2) and the corresponding $D_i[k]$ expressed as (3). From (13), the resulting $D_c[k]$ from the JCF can be expressed as

$$D_c[k] \approx D_i[k] + \hat{\epsilon}[k] F_c(M, \Omega_i) \times 2A_i \cos(k\Omega_i + \phi_i). \quad (19)$$

The residual sampling error after the jitter compensation, defined as $V_e[k] \equiv D_c[k] - V_i(kT_s)$, can be approximated by

$$V_e[k] \approx [\Omega_i + 2F_c(M, \Omega_i)]\, \epsilon[k] \times A_i \cos(k\Omega_i + \phi_i) + 2F_c(M, \Omega_i)\epsilon_e[k] \times A_i \cos(k\Omega_i + \phi_i). \quad (20)$$

The averaged power of $V_e[k]$ is

$$P_e = \frac{A_i^2}{2} [\Omega_i + 2F_c(M, \Omega_i)]^2 \epsilon_{\rm rms}^2 + 2A_i^2 [F_c(M, \Omega_i)]^2 \epsilon_{e,\rm rms}^2 \quad (21)$$

where $\epsilon_{\rm rms}$ is the rms of the clock jitter $\epsilon[k]$, and $\epsilon_{e,\rm rms}$ is the rms of the JDC measurement error $\epsilon_e[k]$. If $A_i = 1$ and the quantization noises are included, the SNR of the signal $D_c[k]$ can be calculated using (4), (5), and (21). From (15), if M is so large that $F_c(M, \Omega_i) \approx -\Omega_i/2$, then the SNR becomes

$$\text{SNR}_{c,\infty} = \frac{1}{(2/3)2^{-2B} + \Omega_i^2 \epsilon_{e,\rm rms}^2}. \quad (22)$$

Comparing (22) with (8), the clock jitter rms $\epsilon_{\rm rms}$ in (8) is replaced by the jitter measurement error rms $\epsilon_{e,\rm rms}$ in (22).

Consider a 16-bit ADC system operating at a sampling rate of 100 MS/s. Its sampling period is $T_s = 10$ ns. Assume that the rms of the clock jitter is $\epsilon_{\rm rms} T_s = 3$ ps, i.e., $\epsilon_{\rm rms} = 3 \times 10^{-4}$. From (22), to ensure an SNR better than 80 dB for an input frequency up to 40 MHz, i.e., $\Omega_i < (4/5)\pi$, one can choose $\epsilon_{e,\rm rms} T_s < 1/\sqrt{12}$ ps and $M = 7$. The $\epsilon_{e,\rm rms}$ indicates the required resolution and accuracy for the JDC. The number of taps for the JCF is $2M + 1 = 15$. The bandwidth limitation is due to the proposed JDC calibration, which will be discussed in Sections V and VI. Fig. 5 shows the SNR performance from simulations of the system in Fig. 1. The JDC is an ideal one with a uniform quantization step size of 1 ps so that $\epsilon_{e,\rm rms} T_s = 1/\sqrt{12}$ ps. The JCF is the one shown in Fig. 4. The circle symbols in Fig. 5 are simulation results. Also shown in the figure are the calculated SNRs of a JCF, with $M = 7$ using (4), (5), and (21). The SNR of an ideal JCF is calculated using (22). At $f_i/f_s = 0.4$ or $\Omega_i = (4/5)\pi$, the JCF can improve the SNR by 20 dB. As a reference, the uncompensated SNR in Fig. 5 is calculated using (8).

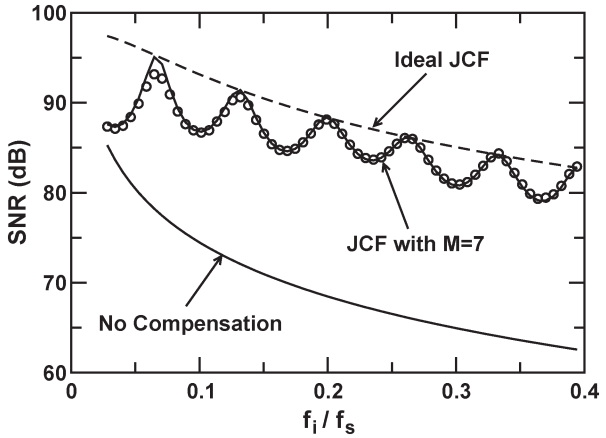

Fig. 5. SNR results of jitter compensation. Circles are from simulations. Lines are from calculations. $f_i = \omega_i/(2\pi)$ is the input frequency, and $f_s = 1/T_s$ is the sampling rate.

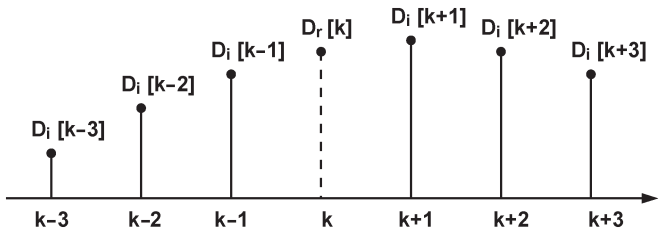

Fig. 6. Signal reconstruction.

III. Signal Reconstruction

Consider a band-limited analog input $V_i(t)$ and the corresponding digital ADC output $D_i[k]$. As shown in Fig. 6, we want to reconstruct the kth sample $D_i[k]$ from its neighboring samples, which can be expressed as $D_i[k-n]$ and $D_i[k+n]$ for integer $n \geq 1$. Since the number of the $D_i[k-n]$ samples is equal to the number of the $D_i[k+n]$ samples, from signal symmetric property, the output of the reconstruction filter can be expressed as

$$D_r[k] = \sum_{n=1}^{N} \{h_r[n] \times (D_i[k-n] + D_i[k+n])\} \quad (23)$$

where $h_r[n]$ are the filter coefficients. Using (10) and neglecting the clock jitter $\epsilon[k]$, the above equation becomes

$$D_r[k] = \sum_{n=1}^{N} \left\{ \frac{1}{2\pi} \int_{-\omega_B}^{+\omega_B} V(j\omega) e^{j\omega k T_s} [2h_r[n] \cos(n\omega T_s)]\, d\omega \right\} = \frac{1}{2\pi} \int_{-\omega_B}^{+\omega_B} V(j\omega) e^{j\omega k T_s} \sum_{n=1}^{N} [2h_r[n] \cos(n\omega T_s)]\, d\omega. \quad (24)$$

For a valid reconstruction filter, the $D_r[k]$ of (24) must be equal to the $D_i[k]$ of (10). Thus, within the frequency band of interest, the $h_r[n]$ coefficients must satisfy

$$\sum_{n=1}^{N} [h_r[n] \cos(n\Omega)] = \frac{1}{2} \quad (25)$$

where $\Omega = \omega T_s$. To find a solution for the $h_r[n]$ coefficients, consider an ideal low-pass filter $X(\Omega)$. For $|\Omega| \leq \Omega_B$,

$X(\Omega) = K$; otherwise, $X(\Omega) = 0$. The Fourier series representation of this filter is

$$X(\Omega) = \frac{K \times \Omega_B}{\pi} + \sum_{n=1}^{\infty} \left[\frac{2K \times \sin(n\Omega_B)}{n\pi} \cos(n\Omega) \right]. \quad (26)$$

Comparing (25) and (26) for N approaching infinity, $h_r[n]$ can be found to be

$$h_r[n] = \frac{\sin(n\Omega_B)}{n(\pi - \Omega_B)}. \quad (27)$$

For example, if the bandwidth of $V_i(t)$ is ω_B, and $\Omega_B = \omega_B T_s = (4/5)\pi$, we can choose

$$h_r[n] = \frac{5 \sin \frac{4n\pi}{5}}{n\pi}. \quad (28)$$

Consider a sine wave input where $V_i(t)$ and $D_i[k]$ are expressed as (2) and (3), respectively. For the reconstruction filter of (23) with a finite value of N and the coefficients of (28), the output is

$$\begin{aligned} D_r[k] \approx & \left[\sum_{n=1}^{N} \frac{10}{\pi} \frac{\cos(n\Omega_i) \sin \frac{4n\pi}{5}}{n} \right] \times A_i \sin(k\Omega_i + \phi_i) \\ & + \frac{5A_i\Omega_i}{\pi} \left\{ \sum_{n=1}^{N} \frac{\cos[\Omega_i(k+n) + \phi_i] \sin \frac{4n\pi}{5}}{n} \epsilon[k+n] \right\} \\ & + \frac{5A_i\Omega_i}{\pi} \left\{ \sum_{n=1}^{N} \frac{\cos[\Omega_i(k-n) + \phi_i] \sin \frac{4n\pi}{5}}{n} \epsilon[k-n] \right\}. \end{aligned} \quad (29)$$

As N increases, it can be shown that the first term on the right-hand side approximates $A_i \sin(k\Omega_i + \phi_i)$, which is the exact value of $V_i(t)$ sampled at $t = kT_s$. The last two terms in (29) are errors caused by the clock jitters when sampling $D_i[k-n]$ and $D_i[k+n]$.

The reconstruction filter will be used in the TDC calibrations described in Sections V and VI.

IV. Stochastic TDC

The JDC in Fig. 1 contains a TDC that digitizes the timing (or phase) difference between two clocks of identical frequency. For a 16-bit 100-MS/s ADC system, the TDC must perform the conversion every clock cycle and must have a resolution better than 1 ps.

There are several different types of TDC architecture. An up-to-date survey can be found in [7]. Most of them require certain types of precision circuits, such as delay elements with precise delay [8], [9], oscillators with precise frequency [7], or time-to-voltage converters with precise conversion function [10], [11]. The TDC used in our design is based on the stochastic TDC architecture [5], [12]. It does not use any precision circuit and can easily be realized in a standard CMOS very large scale integration technology.

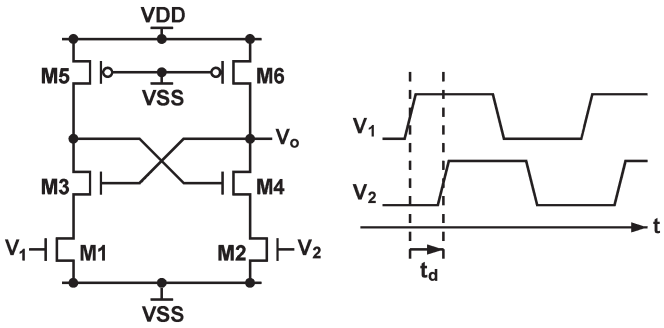

Fig. 7. TCMP.

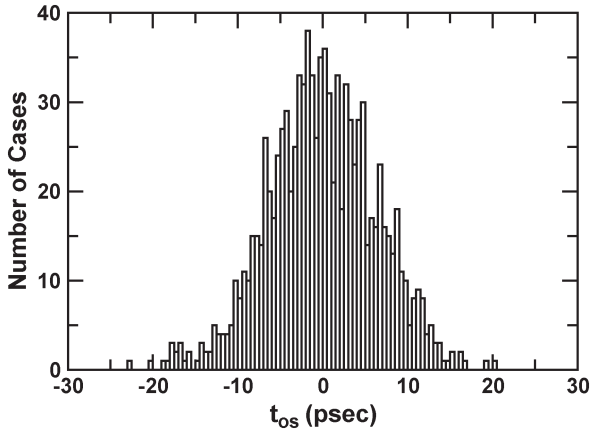

Fig. 8. Histogram of timing comparator offset $t_{\rm os}$, with 1000 Monte Carlo simulations.

A stochastic TDC conducts the time-to-digital conversion by exploring the statistics of a group of timing comparators (TCMPs). Like a flash ADC, it can easily complete the conversion in one clock cycle. It can provide the conversion for every clock cycle continuously. Furthermore, it can improve the conversion resolution simply by adding more TCMPs. Fig. 7 shows a TCMP example [5]. It compares the rising edges of two clocks V_1 and V_2. Ideally, its output is a digital 1 if the timing difference $t_d > 0$; otherwise, the output is a digital 0. However, a practical TCMP exhibits an offset $t_{\rm os}$. The offset is mainly caused by devices mismatches and interconnect mismatches. The TCMP now yields an output of digital 1 only if $t_d > t_{\rm os}$. Fig. 8 shows the $t_{\rm os}$ statistics of a TCMP realized in a 90-nm-CMOS technology. The data were collected from 1000 Monte Carlo circuit simulations. The standard deviation of $t_{\rm os}$ is $\sigma = 6.36$ ps.

Fig. 9 shows the architecture of a stochastic TDC. It contains L TCMPs. Each TCMP detects the polarity of $(t_d - t_{\rm os})$ and has its own $t_{\rm os}$ offset. For every clock cycle, the TDC's output m is generated by summing the digital outputs from all TCMPs. Thus, m is the number of TCMPs with a digital 1 output. Fig. 10 illustrates the TDC transfer function. From the central limit theorem, the probability density function (pdf) of $t_{\rm os}$ is approximately a normal distribution $G(t_{\rm os})$, which is

$$G(t_{\rm os}) = \frac{1}{\sigma\sqrt{2\pi}} e^{-t_{\rm os}^2/(2\sigma^2)} \quad (30)$$

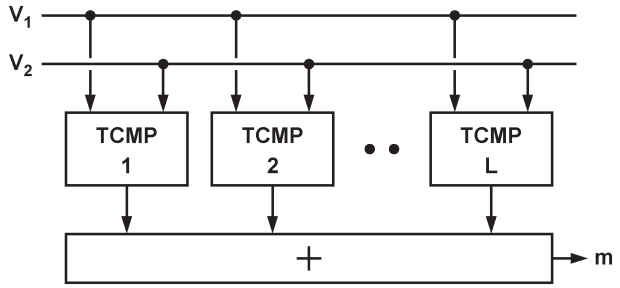

Fig. 9. Stochastic TDC.

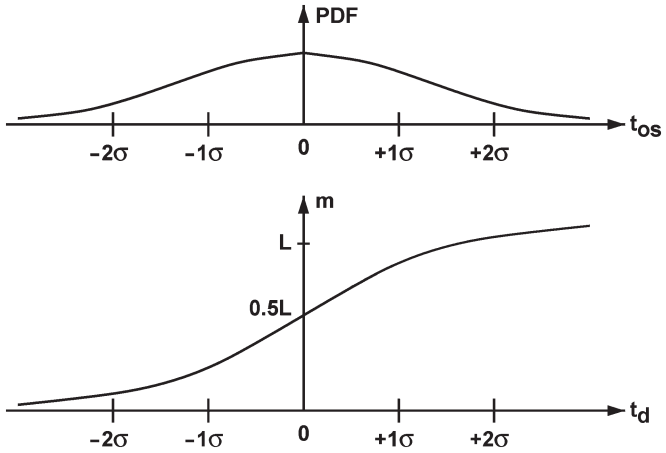

Fig. 10. TDC transfer function and $t_{\rm os}$ pdf.

where σ is the standard deviation of $t_{\rm os}$. The averaged t_d-to-m TDC transfer function can be obtained by integrating over this pdf, i.e.,

$$m = L \times \int_{-\infty}^{t_d} G(t_{\rm os}) dt_{\rm os}. \tag{31}$$

In practice, m can only be an integer between 0 and L. This TDC has its input range proportional to σ. Its resolution is a function of L.

A TDC converts an input t_d into a digital code m. The input range of a TDC is divided into different segments. Each segment is mapped to a different m code. The quantization step size $\Delta t_s(m)$ is the range of the t_d segment mapped to the digital code m. Similar to an ADC, a TDC also introduces quantization noise $\Delta t_q[k]$. A larger $\Delta t_s(m)$ results in a larger $\Delta t_q[k]$. Define $\Delta t_{q,\rm rms}$ as the rms of $\Delta t_q[k]$. Assume an ideal TDC that has identical $\Delta t_s(m)$ for all m. Similar to (5), an ideal TDC that has a uniform quantization step size of $\Delta t_s = 1$ ps gives $\Delta t_{q,\rm rms} = 1/\sqrt{12}$ ps. For a stochastic TDC, the $\Delta t_s(m)$ may vary for different m's. An explicit expression for its $\Delta t_{q,\rm rms}$ is difficult to obtain. Brute-force simulations were used instead. Fig. 11 shows the $\Delta t_{q,\rm rms}$ achieved by 99% of the stochastic TDCs with a given L. The $t_{\rm os}$ of each TCMP in the TDC is randomly chosen. The statistics of $t_{\rm os}$ is the normal distribution of (30). The data are obtained from simulations of 100 000 different TDC cases for each L. The input t_d is assumed to have a normal distribution with a mean of zero and an rms of $\Delta t_{\rm rms}$. Note that depending on applications, the input t_d may have a different statistical distribution. Consider a stochastic TDC

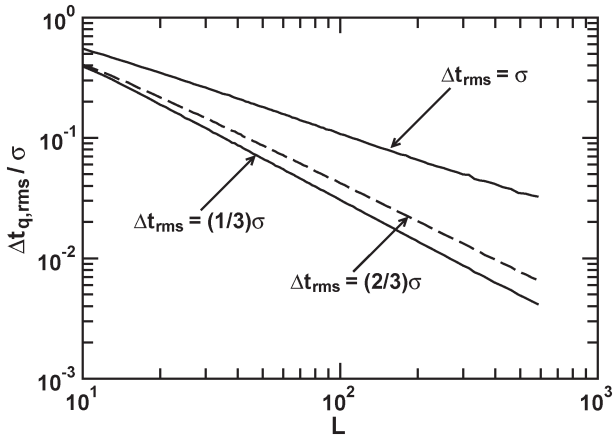

Fig. 11. $\Delta t_{q,\rm rms}$ performance for TDCs with different Ls at various $\Delta t_{\rm rms}$ input levels.

whose internal TCMPs exhibit a $t_{\rm os}$ with standard deviation $\sigma = 6.36$ ps. To achieve the same $\Delta t_{q,\rm rms} = 1/\sqrt{12}$ ps, i.e., $\Delta t_{q,\rm rms}/\sigma = 0.045$, this TDC needs the number of TCMPs $L > 70$ if the rms of the input t_d is $\Delta t_{\rm rms} = (1/3)\sigma$. The TDC needs $L > 94$ if $\Delta t_{\rm rms} = (2/3)\sigma$.

The $t_{\rm os}$ of a TCMP is very sensitive to process, supply voltage, and temperature (PVT) variations. It is also sensitive to the waveforms of the two input clocks V_1 and V_2. Therefore, it is necessary to calibrate a stochastic TDC so that an accurate estimation of t_d can be extracted from its digital output. Techniques to calibrate the stochastic TDC in the background are proposed in the following two sections.

In the following discussion, the rms of the clock jitter, the TDC quantization noise, and its rms are normalized as follows:

$$\epsilon_{\rm rms} = \frac{\Delta t_{\rm rms}}{T_s} \quad \epsilon_q[k] = \frac{\Delta t_q[k]}{T_s} \quad \epsilon_{q,\rm rms} = \frac{\Delta t_{q,\rm rms}}{T_s} \tag{32}$$

where T_s is the nominal clock period. If $\Delta t_{q,\rm rms} = 1/\sqrt{12}$ ps and $T_s = 10$ ns, then $\epsilon_{q,\rm rms} = (1/\sqrt{12}) \times 10^{-4}$.

V. Jitter Compensation Scenario 1

Fig. 12 shows an ADC system with jitter compensation. In this scenario, the external clock $\rm CLK_e$ is assumed to be clean and without jitter. It defines the sampling instants, denoted as kT_s. A variable delay buffer (VDB) receives the $\rm CLK_e$ clock and generates the internal $\rm CLK_i$ clock that drives the ADC. The delay of the VDB is controlled by a delay-control (DCTL) signal such that $\rm CLK_i$ is phase aligned with $\rm CLK_e$. The DCTL signal can be generated from a delay-locked loop or a phase-locked loop. It is assumed that the VDB adds jitter to the $\rm CLK_i$ clock, changing the sampling instants to $(k + \epsilon[k])T_s$. The clock jitter $\epsilon[k]$ is measured and digitized by a JDC. A JCF, similar to the one shown in Fig. 4, uses the measured jitter data to convert the $D_i[k]$ signal from the ADC into the corrected signal $D_c[k]$. The JCF eliminates the sampling error in $D_i[k]$ caused by the jitter $\epsilon[k]$.

The JDC in Fig. 12 consists of a stochastic TDC and a jitter calibration processor JCP1. The TDC measures the timing difference between $\rm CLK_e$ and $\rm CLK_i$. This TDC can also be used

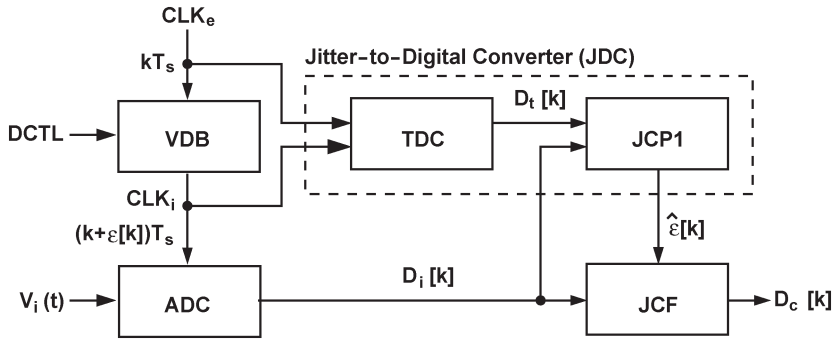

Fig. 12. Jitter compensation scenario 1: clean external clock.

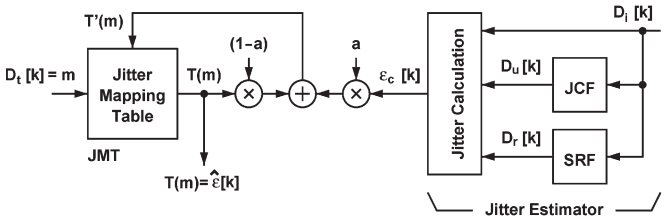

Fig. 13. Block diagram of jitter calibration processor JCP1.

as the phase detector of a digital delay-locked loop that generates the DCTL signal [12]. The JCP1 translates the output $D_t[k]$ from the TDC into the corresponding jitter estimation, $\hat{\epsilon}[k]$. The JCP1 also performs self-calibration to ensure an accurate $\hat{\epsilon}[k]$.

Fig. 13 shows the JCP1 block diagram. It contains a jitter mapping table (JMT). The JMT receives $D_t[k] = m$ as an address and outputs the content stored in that address. The content $T(m)$ is the corresponding jitter estimation $\hat{\epsilon}[k]$. Every address m has its own content $T(m)$. Note that $0 \leq m \leq L$, where L is the number of TCMPs in the TDC. The content $T(m)$ is acquired from a jitter estimator (JE). The JE extracts the jitter information $\epsilon[k]$ from $D_i[k]$ and its neighboring samples, which are digital outputs from the ADC. The JE includes a JCF employing the structure of Fig. 4. This JCF generates the $D_u[k]$ signal as defined in (17). Also included in the JE is a signal reconstruction filter (SRF) that takes $2N$ samples $D_i[k-n]$ and $D_i[k+n]$, where $1 \leq n \leq N$, and applies (23) to reconstruct the $V_i(kT_s)$ sample, denoted as $D_r[k]$. The JE makes a new estimation of the jitter $\epsilon[k]$ based on the outputs from both JCF and SRF. Its output $\epsilon_c[k]$ updates the $T(m)$ by

$$T'(m) = (1-a) \times T(m) + a \times \epsilon_c[k] \tag{33}$$

where $a < 1$ is a constant. The above equation is a low-pass filter with a single pole at $z_p = 1 - a$. Its function is to remove the high-frequency components in $\epsilon_c[k]$ so that $T(m)$ can approximate the dc value of $\epsilon_c[k]$.

The JE calculates its output $\epsilon_c[k]$ based on the criterion described as follows. If a JCF employs this $\epsilon_c[k]$ for jitter compensation, its output $D_c[k]$ should be equal to $D_r[k]$, which is the output of the SRF. From (16), and letting $D_c[k] = D_r[k]$, the $\epsilon_c[k]$ is calculated as

$$\epsilon_c[k] = \frac{D_r[k] - D_i[k]}{D_u[k]} \times \epsilon_u. \tag{34}$$

Consider a sine wave input where $V_i(t)$ and $D_i[k]$ are expressed as (2) and (3), respectively. The filter coefficients for the SRF are the $h_r[n]$ of (28). By equating (19) and (29) and letting $\hat{\epsilon}[k] = \epsilon_c[k]$, we have

$$\epsilon_c[k] \approx -\frac{\Omega_i}{2F_c(M,\Omega_i)} \times \epsilon[k] + \left[\sum_{n=1}^{N}\left(\frac{10}{\pi}\frac{\cos(n\Omega_i)\sin\frac{4n\pi}{5}}{n}\right) - 1\right] \times \frac{\tan(k\Omega_i + \phi_0)}{2F_c(M,\Omega_i)}. \tag{35}$$

Contributions by the last two terms in (29) are not shown in (35). They are removed by the low-pass filter of (33). For each m, the $T(m)$ is updated only when the TDC has its output $D_t[k] = m$. Every time $D_t[k] = m$, the TDC detects a similar $\epsilon[k]$ jitter, denoted as $\epsilon(m)$. From (35), we have

$$T(m) \approx -\frac{\Omega_i}{2F_c(M,\Omega_i)} \times \epsilon(m). \tag{36}$$

The second term on the right-hand side of (35) does not appear in (36) since it is removed by the low-pass filter of (33).

Fig. 14 shows the ratio of $T(m)/\epsilon(m)$ calculated from (36). Note that $\Omega_i = \omega_i T_s = 2\pi f_i / f_s$. The ratio represents the JDC conversion gain. Different M's yield different conversion gain characteristics, where $2M+1$ is the number of taps for the JCF shown in Fig. 13. If $M \to \infty$, the conversion gain approaches 1 for all input frequencies up to $f_i = 0.4 f_s$, which is determined by the $h_r[n]$ coefficients of the SRF. The conversion gain is less accurate at low f_i frequencies. An odd M is preferred if $V_i(t)$ is a narrow-band low-f_i signal. If an even M value is selected, the resulting $|T(m)|$ can be excessively large under the same input condition.

Although a larger M for the JCF results in a better JDC conversion gain characteristic, a large M is not necessary if the JCF in Fig. 13 is identical to the JCF in Fig. 12. If $V_i(t)$ is wideband and has many different frequency components, the averaged JDC conversion gain is approximately 1, even for a small M. To illustrate a narrow-band input condition, let $V_i(t) = A_i \sin(\omega_i t + \phi_i)$ as in (2). From (36), $\hat{\epsilon}[k] = -[\Omega_i/(2F_c(M,\Omega_i))] \times \epsilon[k]$. For jitter compensation, the $D_c[k]$ signal is calculated using (19), resulting in $D_c[k] \approx D_i[k] - A_i\Omega_i \cos(k\Omega_i + \phi_i) \times \epsilon[k]$. The last term on the right-hand

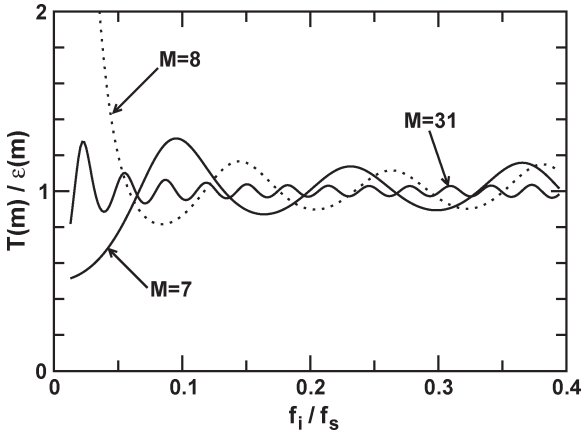

Fig. 14. Conversion gain of the JDC using JCP1.

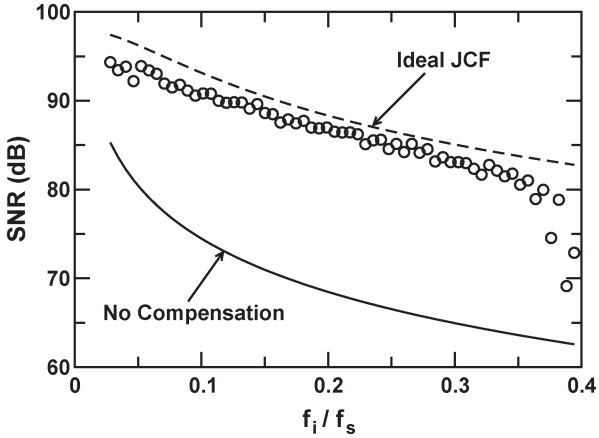

Fig. 15. SNR of the ADC system of Fig. 12 at various input frequencies.

side is an accurate estimation of the sampling error in $D_i[k]$, which is shown in (6). Thus, the JDC conversion gain error is automatically compensated. It has little effect on the SNR of the $D_c[k]$ signal.

For an accurate signal reconstruction, the SRF requires a very large N. It is not practical to implement the SRF hardware with $2N$ multipliers. We propose using the cyclic multiply-and-accumulate (MAC) architecture to realize the SRF. The MAC architecture requires only one multiplier and one accumulator. As a result, the SRF can generate only one $D_r[k]$ signal every $2N+1$ clock cycles. Therefore, the proposed JE generates one valid $\epsilon_c[k]$ output every $2N+1$ clock cycles.

For the jitter compensation configuration of Fig. 12, the jitter measurement error of the JDC, defined as $\epsilon_e[k]$ in (1), is mainly caused by the quantization noise $\epsilon_q[k]$ of its TDC. We can assume $\epsilon_e[k] = \epsilon_q[k]$ and $\epsilon_{e,\mathrm{rms}} = \epsilon_{q,\mathrm{rms}}$. The JDC's $\epsilon_{e,\mathrm{rms}}$ requirement can be estimated by using (4) and (21). If M is large, it can be found by using (22). Since $\epsilon_{e,\mathrm{rms}} = \epsilon_{q,\mathrm{rms}}$, we also obtain the $\epsilon_{q,\mathrm{rms}}$ requirement for the TDC, which dictates the TDC's resolution.

The ADC system of Fig. 12 is simulated using a C program. As in Section II, the ADC has 16-bit resolution and operates at a sampling rate of 100 MS/s. The rms of the clock jitter $\epsilon[k]$ is $\epsilon_{\mathrm{rms}} = 3 \times 10^{-4}$, i.e., $\Delta t_{\mathrm{rms}} = \epsilon_{rm} T_s = 3$ ps. To make simulation results agree with the theoretical analyses more closely, the TCMPs in the TDC are assumed to contain uniformly distributed t_{os} so that the TDC has a uniform quantization step of size $\Delta t_s = 1$ ps. The resulting rms of quantization noise is $\epsilon_{q,\mathrm{rms}} = (1/\sqrt{12}) \times 10^{-4}$. In practice, 72 TCMPs are required in the TDC to achieve the same $\epsilon_{q,\mathrm{rms}}$ if the TCMP's t_{os} has the normal distribution $G(t_{\mathrm{os}})$ of (30) with a standard deviation $\sigma = 6.36$ ps. For the JCF, $M = 7$. For the SRF, $N = 2^{10}$ and $\Omega_B = (4/5)\pi$. For the low-pass filter of (33), $a = 2^{-13}$. In Fig. 15, the circles are the simulation results. They are the SNRs of the $D_c[k]$ signal under various f_i frequencies. From (22) with $\epsilon_{e,\mathrm{rms}}$ replaced by $\epsilon_{q,\mathrm{rms}}$, the theoretical SNR with ideal JCF compensation is shown as the dash line. From (8), the theoretical SNR for the ADC without jitter compensation is shown as the solid line. The calibration scheme of Fig. 12 improves the SNR by about 20 dB.

VI. Jitter Compensation Scenario 2

The ADC system of Fig. 12 assumes a clean external clock, which is used as a reference to measure the jitter caused by the internal clock buffer. Fig. 16 shows an alternative ADC system with the external clock exhibiting $\epsilon[k]$ jitter. The system can correct the sampling errors caused by the clock jitter $\epsilon[k]$, thus allowing the use of cheaper clock source. However, jitter compensation for this scenario is more complex. Unlike the system of Fig. 12, there is no clean clock that the TDC can refer to for measuring the absolute jitter. What can be measured is the relative jitter between two consecutive sampling instants [11]. In Fig. 16, the CLK clock dictates the time of kth sampling at $(k + \epsilon[k])T_s$, where T_s is the nominal sampling interval and $\epsilon[k]$ is the absolute jitter normalized to T_s. The CLK$_d$ clock is the CLK delayed by one T_s. The sampling time provided by CLK$_d$ is denoted as $(k + \epsilon[k-1])T_s$. Fig. 17 shows the relationship between the two clocks. Sampling instants are at the rising edges of the clocks. The TDC in Fig. 16 measures the time difference between the kth edge of the CLK clock and the $(k-1)$th edge of the CLK$_d$ clock. The measurement result is the cycle jitter defined as

$$\tau[k] = \epsilon[k] - \epsilon[k-1]. \tag{37}$$

The delay between the clock CLK and the clock CLK$_d$ should not introduce additional jitter. It can be realized using a passive delay line. For the calibration scheme described below, the accuracy of the realized delay is not crucial as long as it is constant. If the realized delay is different from T_s, the difference becomes a constant timing shift between the two input clocks of the TDC. This effect can be treated as a dc offset of the TDC. In the proposed scheme, the realized delay is only required to be close to T_s so that the offset is small enough to avoid overloading the TDC.

In Fig. 16, a TDC converts the cycle jitter $\tau[k]$ into a $D_t[k]$ sequence. An ADC digitizes the analog input $V_i(t)$ and generates the corresponding digital output $D_i[k]$. A jitter calibration processor JCP2 generates the absolute jitter estimation $\hat{\epsilon}[k]$. A JCF then uses this $\hat{\epsilon}[k]$ to correct $D_i[k]$. The corrected output is $D_c[k]$. Fig. 18 shows the block diagram of the proposed JCP2. A JMT receives $D_t[k] = m$ from the TDC as an address and

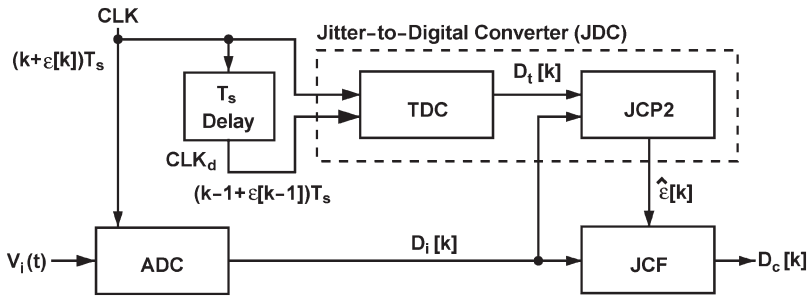

Fig. 16. Jitter compensation scenario 2: jittering external clock.

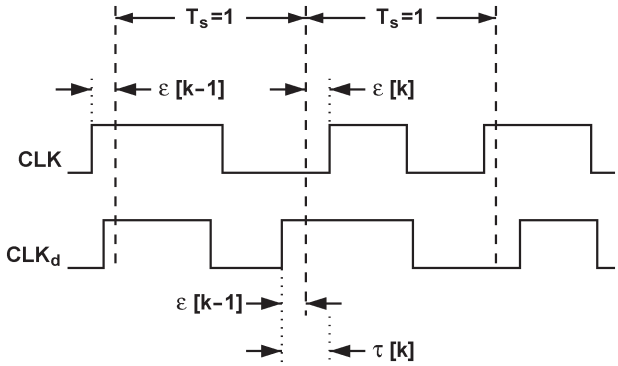

Fig. 17. Relations between the CLK and CLK$_d$ clocks.

outputs the content stored in that address. The content $T(m)$ is an estimation of the cycle jitter defined in (37), denoted as $\hat{\tau}[k]$. The absolute jitter $\hat{\epsilon}[k]$ is obtained by lossy accumulation of the $\hat{\tau}[k]$ sequence. The operation is expressed as

$$\hat{\epsilon}[k] = \hat{\tau}[k] + b \times \hat{\epsilon}[k-1] \tag{38}$$

where $b < 1$ is a constant. The reason to use a lossy accumulator is that a dc component may appear in the quantization noise of the TDC, which can overflow a lossless accumulator.

To calibrate the content $T(m)$ when $D_t[k] = m$, JCP2 includes two JEs, namely, JE1 and JE2. Both are identical to the one shown in Fig. 13. JE1 collects $D_i[k]$ and its neighboring $2N$ samples to extract $\epsilon_c[k]$, which is an estimation of the absolute jitter $\epsilon[k]$. In addition, JE2 collects $D_i[k-1]$ and its neighboring $2N$ samples to extract $\epsilon_c[k-1]$. Two jitter estimators are required since neither estimator can calculate both $\epsilon_c[k]$ and $\epsilon_c[k-1]$ consecutively. Each estimator can generate only one valid output for every $2N+1$ samples. Similar to (37), we define $\tau_c[k]$ as

$$\tau_c[k] = \epsilon_c[k] - \epsilon_c[k-1]. \tag{39}$$

In Fig. 18, the content $T(m)$ is updated by applying the low-pass-filter function of (33) on $\tau_c[k]$ so that $T(m)$ can approximate the mean value of $\tau_c[k]$. Note that $T(m)$ only records the estimation of the cycle jitter $\tau[k]$ when $D_t[k] = m$. This cycle jitter is denoted as $\tau(m)$. The TDC's dc offset may affect the value of m, but it has no effect on the accuracy of $T(m)$.

Combining (39) and (35), neglecting the high-frequency components that can be removed by the low-pass filter of (33), $\tau_c[k]$ can be approximated by

$$\begin{aligned}\tau_c[k] \approx & -\frac{\Omega_i}{2F_c(M,\Omega_i)} \times \tau[k] - \frac{5\Omega_i \cos\Omega_i}{2\pi F_c(M,\Omega_i)} \sin\left(\frac{4\pi}{5}\right) \\ & \times \tau[k] + \left\{\left[\sum_{n=1}^{N} \frac{10}{\pi} \frac{\cos(n\Omega_i) \times \sin\frac{4n\pi}{5}}{n}\right] - 1\right\} \\ & \times \frac{\tan(k\Omega_i + \phi_i) - \tan\left[(k-1)\Omega_i + \phi_i\right]}{2F_c(M,\Omega_i)}. \end{aligned} \tag{40}$$

The first term on the right-hand side of (40) is the desired jitter estimation, which approaches $\tau[k]$ for large M. The third term on the right-hand side has a nonzero mean. However, it is significant only when Ω_i is close to $\Omega_B = (4/5)\pi$ or $\Omega_B = 0$.

The second term on the right-hand side of (40) demands special attention. It originates from the fact that, for any specific m, the sampling interval between the reconstructed $D_r[k]$ in JE1 and the reconstructed $D_r[k-1]$ in JE2 is always $T_s + \tau(m)$, as illustrated in Fig. 19. To solve this constant $\tau(m)$ issue, the $D_i[k]$ sample used in the SRF of JE2 is replaced by an interpolated $D_i'[k]$, which has its sampling time moved backward by $\tau(m)$. The $D_i'[k]$ can be calculated by using

$$D_i'[k] = D_i[k] + \frac{T(m)}{\epsilon_u} \times D_u[k] \tag{41}$$

where $D_u[k]$ and ϵ_u come from the JCF shown in Fig. 16. When the calibration process converges, the resulting $T(m)$ can be expressed as

$$T(m) = \frac{-\pi\Omega_i - 5\Omega_i \cos\Omega_i \times \sin\left(\frac{4\pi}{5}\right)}{2\pi F_c(M,\Omega_i) - 5\Omega_i \cos\Omega_i \times \sin\left(\frac{4\pi}{5}\right)} \times \tau(m). \tag{42}$$

Fig. 20 shows the ratio of $T(m)/\tau(m)$ of (42). For $M \to \infty$, the conversion gain approaches 1 for all input frequencies up to $f_i = 0.4 f_s$.

In Fig. 18, the final absolute jitter estimate $\hat{\epsilon}[k]$ is obtained by the lossy accumulation of $\hat{\tau}[k]$, as expressed in (38). Assuming $T(m) = \tau(m)$, from (1), (37), and (38), the rms of the jitter estimation error can be found as

$$\epsilon_{e,\mathrm{rms}}^2 = \frac{1-b}{1+b} \times \epsilon_{\mathrm{rms}}^2 + \frac{1}{1-b^2} \times \epsilon_{q,\mathrm{rms}}^2. \tag{43}$$

For an accurate estimation of the jitter $\epsilon[k]$, it is necessary to choose the b coefficient close to 1. However, as b approaches 1,

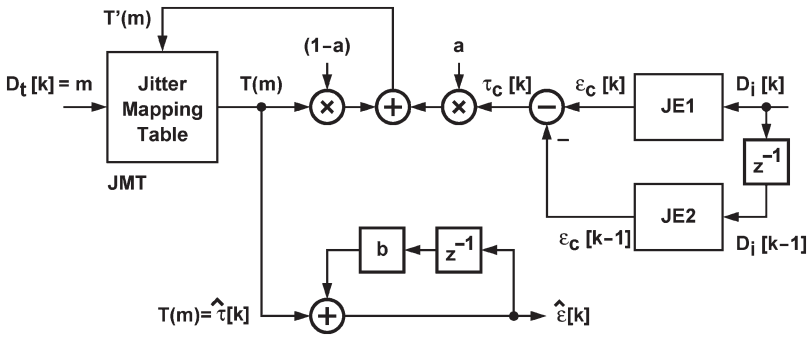

Fig. 18. Block diagram of jitter calibration processor JCP2.

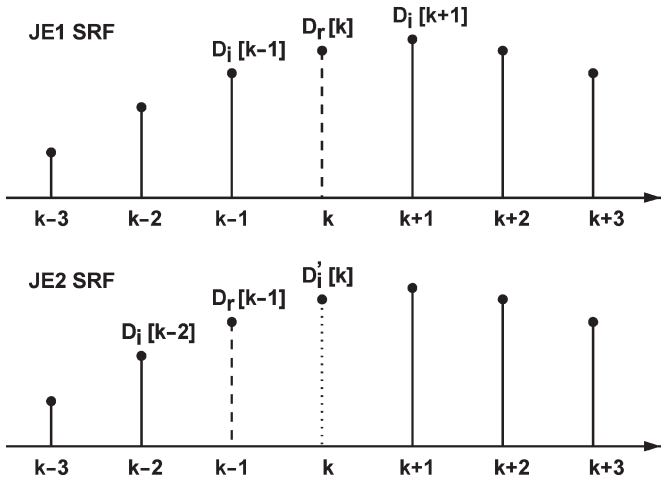

Fig. 19. Signal reconstruction in JE1 and JE2.

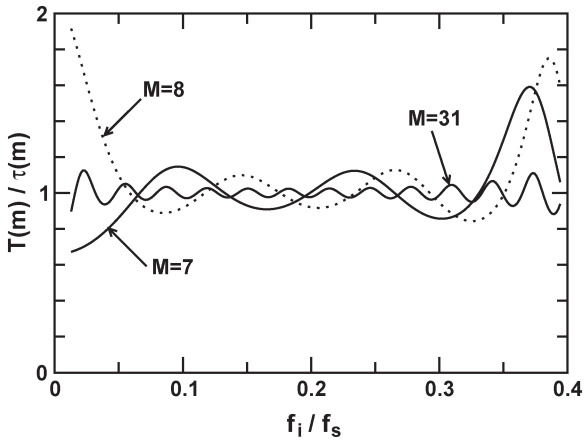

Fig. 20. Conversion gain of the JDC using JCP2.

the accumulation of the low-frequency components in ϵ_q can become the major source of $\epsilon_e[k]$. For a given $\epsilon_{\rm rms}$ and $\epsilon_{q,\rm rms}$, the optimal value of b for a minimum $\epsilon_{e,\rm rms}$ is

$$b = \frac{2\epsilon_{\rm rms}^2 + \epsilon_{q,\rm rms}^2 - \sqrt{\epsilon_{q,\rm rms}^4 + 4\epsilon_{\rm rms}^2\epsilon_{q,\rm rms}^2}}{2\epsilon_{\rm rms}^2}. \quad (44)$$

The ADC system of Fig. 16 is simulated using a C program. System and design parameters are identical to the ADC described in Section V. Since the rms of jitter $\epsilon[k]$ is $\epsilon_{\rm rms} = 3 \times 10^{-4}$, and the rms of the TDC quantization noise is $\epsilon_{q,\rm rms} = (1/\sqrt{12}) \times 10^{-4}$, an optimal $b = 0.9$ is chosen for the lossy accumulation of $\hat{\tau}[k]$. The circle symbols in Fig. 21 are the simulation results. They are the SNRs of the $D_c[k]$ signal under various f_i frequencies. The proposed jitter compensation can improve the SNR by about 10 dB. The theoretical SNR by assuming an ideal JCF for jitter compensation is shown as the dash line. It is calculated by (22) with $\epsilon_{e,\rm rms}$ of (43).

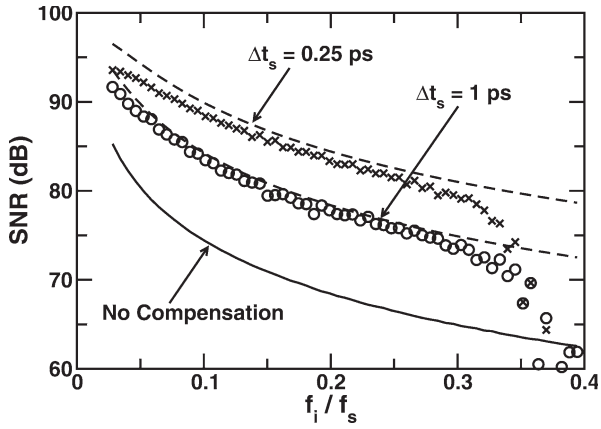

Fig. 21. Simulated SNRs of the ADC system of Fig. 16.

The SNR can be improved by increasing the resolution of the TDC. Consider a TDC with a uniform quantization step of size reduced to $\Delta t_s = 0.25$ ps. Its rms of quantization noise then becomes $\epsilon_{q,\rm rms} = (0.25/\sqrt{12}) \times 10^{-4}$. The corresponding optimal value for b is 0.97. The simulation results are shown as the cross symbols in Fig. 21. The jitter compensation can improve the SNR by about 16 dB. However, to achieve an equivalent rms of quantization noise $\epsilon_{q,\rm rms} = (0.25/\sqrt{12}) \times 10^{-4}$, the TDC must contain $L = 236$ TCMPs, if the TCMPs have a $t_{\rm os}$ with standard deviation $\sigma = 6.36$ ps.

VII. Implementation Issues

Consider a 16-bit 100-MS/s ADC. To relax the clock jitter requirement, it employs the jitter compensation system shown in Fig. 16. For this system, a passive delay line can be used to provide the $T_s = 10$ ns delay to generate the $\rm CLK_d$ clock. Functional blocks, such as TDC, JCP2, and JCF, can be realized in a standard CMOS technology. They can also be integrated with the ADC on the same chip.

Assume the jitter compensation system is realized in a 90-nm-CMOS technology. The TDC consists of $L = 127$ TCMPs. The TCMP shown in Fig. 7 consumes 26.7 μW of power at a clock rate of 100 MHz. Thus, the TDC consumes a total power of about 3.39 mW. Its chip area is estimated to be 1000 μm^2. Note that the resolution and input range of the TDC depend on the PVT variation of the TCMPs. Comprehensive characterization of the fabrication technology is required for optimized design of the TDC.

The JCF is a digital FIR filter, as shown in Fig. 4. If $M = 7$, it includes 200 flip-flops, 15 multipliers, and 14 adders. Although the $D_i[k]$ signal is 16 bits wide, a width 8 bits is sufficient when calculating the $D_u[k]$ signal. The $h_s[n]$ coefficients are 6 bits wide. Synthesized by computer-aided-design (CAD) software, this JCF consumes 2.35 mW of power and occupies an area of 17 000 μm^2.

The JCP2 is also a digital functional block. It contains a JMT with 128 entries. Each entry is 23 bits wide. The JCP2 shown in Fig. 18 includes two JEs, i.e., JE1 and JE2, for generating $\epsilon_c[k]$ and $\epsilon_c[k-1]$. The two JCFs in JE1 and JE2 and the JCF in Fig. 16 can share the same hardware. Each JE contains an SRF. Although the SRF is a FIR filter, to save power and area, it is realized using the MAC architecture. Each SRF consists of a multiplier and an accumulator. The two SRFs share a ROM that stores the filter coefficients $h_r[n]$. The ROM has $N = 2^{10}$ entries, and each entry is 18 bits wide. Due to the MAC operation, each SRF produces only one valid output every $2N + 1$ clock cycles. Synthesized by CAD software, this JCP2 consumes 4.42 mW of power and occupies an area of 185 000 μm^2. Note that the JMT occupies about 58% of the JCP2's total area. The area and power can be reduced by customized design of the JMT.

Realized in a 90-nm-CMOS technology, this jitter compensation system consumes a total power of about 10 mW.

VIII. Conclusion

High-performance Nyquist-rate ADCs require low-jitter clocks for input sampling. In this paper, we have proposed digital signal processing techniques to relax the clock jitter requirement. Enabling techniques are the stochastic TDC and the JCF. The TDC comprises a large group of TCMPs and relies on the statistic variation of the TCMPs for time-to-digital conversion. The TDC is sensitive to the PVT variations. It requires calibration for accurate measurement. Our proposed calibration schemes are based on signal reconstruction. They can be performed in the background without interrupting the normal ADC operation.

All the signal processing described in this paper can be realized using digital circuitry. The timing comparators in the TDC are simple latches and can be realized in any standard CMOS technology. Thus, both the chip area and power consumption of the proposed techniques are expected to be reduced as CMOS technologies advance.

References

[1] M. Shinagawa, Y. Akazawa, and T. Wakimoto, "Jitter analysis of high-speed sampling systems," *IEEE J. Solid-State Circuits*, vol. 25, no. 1, pp. 220–224, Feb. 1990.

[2] H. Kobayashi, M. Morimura, K. Kobayashi, and Y. Onaya, "Aperture jitter effects in wideband sampling systems," in *Proc. IEEE Instrum. Meas. Technol. Conf.*, May 1999, vol. 2, pp. 880–885.

[3] B. Brannon and R. Reeder, *Understanding High Speed ADC Testing and Evaluation*. Norwood, MA: Analog Devices, Inc., Application Note AN-835, Apr. 2006.

[4] J. Tourabaly and A. Osseiran, "A jittered-sampling correction technique for ADCs," in *IEEE Int. Symp. Electron. Des., Test Appl.*, Jan. 2008, pp. 249–252.

[5] V. Gutnik and A. Chandrakasan, "On-chip picosecond time measurement," in *Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2000, pp. 52–53.

[6] M. Gustavsson, J. J. Wikner, and N. N. Tan, *CMOS Data Converters for Communications*. Boston, MA: Kluwer, 2000.

[7] A. K. M. K. Mollah, R. Rosales, S. Tabatabaei, J. Cicalo, and A. Ivanov, "Design of a tunable differential ring oscillator with short start-up and switching transients," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 12, pp. 2669–2682, Dec. 2007.

[8] P. Dudek, S. Szczepañski, and J. V. Hatfield, "A high-resolution CMOS time-to-digital converter utilizing a Vernier delay line," *IEEE J. Solid-State Circuits*, vol. 35, no. 2, pp. 240–247, Feb. 2000.

[9] J.-P. Jansson, A. Mäntyniemi, and J. Kostamovaara, "A CMOS time-to-digital converter with better than 10 ps single-shot precision," *IEEE J. Solid-State Circuits*, vol. 41, no. 6, pp. 1286–1296, Jun. 2006.

[10] T. Xia and J.-C. Lo, "Time-to-voltage converter for on-chip jitter measurement," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 6, pp. 1738–1748, Dec. 2003.

[11] M. Ishida, K. Ichiyama, T. J. Yamaguchi, M. Soma, M. Suda, T. Okayasu, D. Watanabe, and K. Yamamoto, "A programmable on-chip picosecond jitter measurement circuit without a reference-clock input," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2005, pp. 512–513.

[12] V. Kratyuk, P. K. Hanumolu, K. Ok, K. Mayaram, and U.-K. Moon, "A digital PLL with a stochastic time-to-digital converter," in *Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2006, pp. 38–39.

Chi-Wei Fan (S'06) was born in Hsinchu, Taiwan. He received the B.S. degree in electrical engineering from the National Central University, Chung Li, Taiwan, in 2000. He is currently working toward the Ph.D. degree in electronics engineering with the National Chiao-Tung University, Hsinchu.

His research interests include mixed-signal high-speed high-resolution integrated circuits design for data communications.

Jieh-Tsorng Wu (S'83–M'87–SM'06) was born in Taipei, Taiwan. He received the B.S. degree in electronics engineering from the National Chiao-Tung University, Hsinchu, Taiwan, in 1980, and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1983 and 1988, respectively.

From 1980 to 1982, he served in the Chinese Army as a Radar Technical Officer. From 1982 to 1988, he was with Stanford University, working on high-speed analog-to-digital conversion in CMOS VLSI. From 1988 to 1992, he was with the Hewlett-Packard Microwave Semiconductor Division, San Jose, CA, as a Member of Technical Staff, responsible for several linear and digital gigahertz IC designs. Since 1992, he has been with the Department of Electronics Engineering, National Chiao-Tung University, where he is currently a Professor. His current research interests include high-performance mixed-signal integrated circuits.

Dr. Wu is a member of Phi Tau Phi.