Improved Blind Timing Skew Estimation Based on Spectrum Sparsity and ApFFT in Time-Interleaved ADCs

Sujuan Liu, *Member, IEEE,* Ning Lyu, Jiashuai Cui, and Yuexian Zou, *Senior Member, IEEE*

***Abstract*— Timing skews among channels degrade seriously the time-interleaved analog-to-digital converter (TIADC) performance, which can be improved by the blind timing skew estimation (TSE) technique. In this paper, we proposed the all-phase fast Fourier transform (ApFFT) based on spectrum sparsity signal phase relationship blind TSE (ApFFT-SSPR-BLTSE) algorithm. The ApFFT-SSPR-BLTSE algorithm reduces computational complexity based on the phase relationship of the total output from TIADC and the corresponding reference channel output compared with the existing spectrum sparsity blind TSE (SS-BLTSE) algorithm. We also utilized the ApFFT technique to increase the accuracy of phase spectral estimation. Simulation results show that the proposed ApFFT-SSPR-BLTSE algorithm, which as a reduced number of fast Fourier transforms (FFTs) and low hardware complexity, has higher accuracy for blind TSE compared to the existing SS-BLTSE algorithm. In addition, this paper presents an efficient hardware architecture of the ApFFT-SSPR-BLTSE algorithm on the Xilinx Virtex-6 vlx550tff1759 field-programmable gate array (FPGA) chip for the blind TSE of the four-channel 400-MHz 14-bit TIADC real system. The validation results show that the proposed algorithm uses only a few percent of the hardware resources of the FPGA chip, and the mismatch spurs were suppressed to better than −81.54 dB.**

***Index Terms*— All-phase fast Fourier Transform (ApFFT), nonoverlapping point, spectrum sparsity, time-interleaved analog-to-digital converter (TIADC), timing skews**

I. Introduction

TO MEET the demands of digital signal processing and high performance of mixed-signal systems, high-speed and high-resolution analog-to-digital converters (ADCs) are required. It is complicated to achieve a high speed and high resolution simultaneously for a single state-of-the-art complementary metal–oxide–semiconductor ADC. The time-interleaved analog-to-digital converter (TIADC) structure is a good technique for increasing significantly the sampling rate by using M identical parallel ADCs (sub-ADCs), as shown in Fig. 1 [1]. Ideally, all M parallel channels are assumed to have the same gain and offset and should be operating at precise equally spaced sampling time instants. Therefore, the overall sampling rate of TIADC is M times the sampling rate of a single ADC.

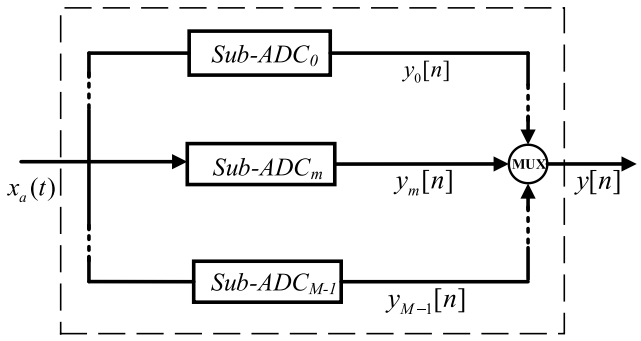


Fig. 1. TIADC system.

Due to practical implementation constraints, some channel mismatches, such as gain mismatch, offset mismatch, and timing skew mismatch, in the TIADC system are inevitable [2]. It is generally known that channel mismatches will result in distortion of sampled waveforms and reduce the spurious-free dynamic range (SFDR) of the TIADC [2]–[4]. In addition, digital correction of frequency-dependent mismatch error is also essential for high-resolution and high-speed TIADCs [5]. Among these mismatches, gain and offset mismatches can be easily calibrated and timing skew mismatch is difficult to calibrate. Timing skew mismatch compensation has attracted substantial attention recently. However, most of the traditional algorithms require an accurate estimate of the timing skew mismatch.

Blind estimation techniques in [6]–[8] do not require knowledge of the input signal to estimate the channel mismatch parameters, and the timing mismatch compensation performance is inferior to those of the nonblind mismatch compensation techniques in [9]–[11]. A multichannel-filtering method is utilized in [9] for TIADC mismatch compensation, which has lower computational complexity and is more robust than the filter-bank approach. An adaptive calibration method is proposed in [12], which is a training method that is suitable for high-resolution applications since it can correct general linear mismatches at the expense of system suspension during

Manuscript received August 31, 2018; revised February 1, 2018; accepted April 13, 2018. This work was supported in part by the Beijing Natural Science Foundation under Grant 4162014 and in part by the Scientific Research Project of Beijing Educational Committee under Grant KM201810005022. The Associate Editor coordinating the review process was Dr. Branislav Djokic. *(Corresponding author: Yuexian Zou.)*

S. Liu, N. Lyu, and J. Cui are with the College of Microelectronics, Beijing University of Technology, Beijing 100124, China (e-mail: liusujuan@bjut.edu.cn; lvningss@163.com; cjsh2013@emails.bjut.edu.cn).

Y. Zou is with the College of Information Engineering, Peking University, Shenzhen 518055, China (e-mail: zouyx@pkusz.edu.cn).

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

Digital Object Identifier 10.1109/TIM.2018.2834080

each calibration. The experimental results showed that these nonblind compensation methods all require accurate measurements of the channel parameters.

In contrast, the blind timing skew estimation (TSE) method is more favorable because it can track channel mismatch variations without interrupting the work of the TIADC system and operate in an online manner. However, several studies [13]–[19] indicate that the blind TSE method may have some limitations.

1) In [14], the mismatches are compensated using a proposed iterative stochastic gradient minimization algorithm. However, this algorithm is of high computational complexity.
2) Jamal *et al.* [16] and Huang and Levy [19] utilized the alias-free bandwidth to estimate the channel mismatch parameters in a blind estimation manner. However, the approach in [16] is only suitable for two-channel TIADC.
3) The oversampling of input signals is necessary in [17].
4) Zou *et al.* [18] proposed an efficient blind estimation method that evaluates the autocorrelation of the analog input signal and the mean squared difference.

However, the approach only performs well when the frequency bandwidth of the analog input signal is lower than half of the Nyquist frequency. In [1], the proposed spectrum sparsity blind TSE (SS-BLTSE) algorithm performs well and can estimate accurately the timing skew when the input analog signal has a specific frequency spectral sparsity attribute. There is a tradeoff between its computational cost and TSE accuracy.

In this paper, we focus on the blind TSE method of TIADC system and estimate the channel timing skews in a blind manner, where the channel gain mismatch and offset mismatch are not taken into account. We proposed the all-phase fast Fourier transform (ApFFT) based on spectrum sparsity signal phase relationship blind TSE (ApFFT-SSPR-BLTSE) algorithm, which we developed by improving the SS-BLTSE algorithm in [1] based on the phase relationship of the spectrum sparsity signal at the nonoverlapping frequency points between the total output from TIADC and its corresponding reference channel output. The spectrum sparsity is defined comprehensively and two types of amplitude spectrum of the analog input signals are discussed: the range continuous spectrum and the discrete spectrum in [1]. As discussed in [1], the spectrum sparsity of the input signal is defined as $S_{\mathrm{SL}} = N_k/N_T$, where N_k denotes the number of nonzero spectrum components and N_T denotes the total number of the spectrum components. In this paper, we use following definitions from [1]: 1) the input signal is sparse when S_{SL} is smaller than $2/M$ for the range continuous spectrum and 2) the input signal may be sparse and the nonoverlapping frequency may be nonzero when $S_{\mathrm{SL}} < 2/M$ for the discrete spectrum. In conclusion, the smaller S_{SL} is the better our ApFFT-SSPR-BLTSE algorithm performs. We also utilized the ApFFT technique to estimate the phase spectrum accurately. Simulation results show that the improved ApFFT-SSPR-BLTSE algorithm can reduce the number of fast Fourier transform (FFT) operations with low computational complexity for hardware implementation. In addition, the improved ApFFT-SSPR-BLTSE algorithm has higher accuracy in the blind TSE compared to the existing SS-BLTSE algorithm. In this paper, an efficient estimated system hardware architecture of the ApFFT-SSPR-BLTSE algorithm is presented for the blind TSE of the TIADC. The proposed architecture is implemented on the Xilinx Virtex-6 vlx550tff1759 field-programmable gate array (FPGA). This synthesized design consumes only a few percent of the hardware resources of the FPGA chip with a clock frequency of 116.82 MHZ.

In this paper, we use the following assumptions and definitions in developing the blind TSE algorithm [1].

1) Input signal $x_a(t)$ is bandlimited with the highest frequency smaller than $f_s/2$.
2) The system parameters of the TIADC system are known, such as the number of sub-ADCs (M) and the overall sampling frequency (f_s).
3) The discrete output signals ($y_m(n)$, $y(n)$) are available.
4) No channel dc offset and gain mismatches are considered since they can be compensated using the simple subtraction and scalar gain approaches.
5) No additive noise is considered.
6) As the timing skew is usually much smaller than the sampling period (smaller than 10%), the input analog signal is much stronger than the disturbance that is introduced by TIADC channel timing skews.
7) The first ADC channel is used as the reference channel when estimating the timing skew of the TIADC system, namely, $\Delta t_0 = 0$.

A literature review shows that the above assumptions are commonly satisfied in TIADC technology research.

This paper is organized as follows. The system model of TIADC, a new TSE model, and the motivation for the new model are presented in Section II. The ApFFT-SSPR-BLTSE algorithm is explained in detail and summarized in Section III. Simulation results are presented in Section IV. The hardware implementation of the ApFFT-SSPR-BLTSE algorithm is described in Section V. FPGA implementation results are presented in Section VI, and our conclusions are discussed in Section VII.

II. System Model and Estimation Modeling

In this section, we will introduce the system model of TIADC, which was introduced in [1] and propose a TSE model.

A. System Model of TIADC

The architecture of the TIADC system is shown in Fig. 1, where sub-ADCs operate in the under-sampling situation with an under-sampling factor M relative to the overall TIADC system. The relation among $x_a(t)$, $y(n)$, and $y_m(n)$ of the TIADC in the time domain and the spectrum domain are described in [1].

From Fig. 1, the analog input $x_a(t)$ is a bandlimited signal and the discrete-time-domain expression of the output of the m_{th} sub-ADC with a sampling period T_{ss} is obtained

$$y_m(n) = x_a(t)|_{t=nT_{\mathrm{ss}}+mT_s+t_m} \tag{1}$$

where $n = 0, 1, \ldots, N/M - 1, m = 0, 1, \ldots, M - 1, T_s$ represents the sampling period of the TIADC system, and Δt_m represents the timing skews for the m_{th} sub-ADC. N is the total number of samples for TIADC and m is the channel index. The digital spectrum [the discrete-time Fourier transform (DTFT)] of $y_m(n)$ is denoted by $Y_m(e^{j\varpi})$ and expressed as follows:

$$Y_m(e^{j\varpi}) = \frac{1}{T_{\text{ss}}} \sum_{k=-\infty}^{+\infty} X_a\left(e^{j\left(\Omega - \frac{2\pi k}{T_{\text{ss}}}\right)}\right) e^{j\left(\frac{\varpi}{T_{\text{ss}}} - \frac{2\pi k}{T_{\text{ss}}}\right)\left(m + \Delta t_m\right) T_s} \tag{2}$$

where ϖ and ω are the digital angular frequencies of the single-channel output and total output signals, respectively. The digital frequency ϖ that is associated with $f_{\text{ss}} = 1/T_{\text{ss}}$ has the following relation with the analog frequency Ω. According to (2), $Y_m(e^{j\varpi})$ is the sum of the frequency-scaled and the shifted versions of $X_a(j\Omega)$, where $X_a(j\Omega)$ is the Fourier transform of input signal $x_a(t)$. In addition, $Y_m(e^{j\varpi})$ is also a periodic function of ϖ with period 2π.

According to assumption 1), the DTFT of $y_0(n)$ can be expressed as follows:

$$\begin{aligned} Y_0(e^{j\varpi}) &= \frac{1}{T_{\text{ss}}} \sum_{k=N_h - M + 1}^{N_h} X_a\left(e^{j\left(\Omega - \frac{2\pi k}{T_{\text{ss}}}\right)}\right) \\ &= \frac{1}{T_{\text{ss}}} \sum_{k=0}^{M-1} X_a\left(e^{j\Omega^{(-k)}_{<2\pi f_s>}}\right) \end{aligned} \tag{3}$$

where we define $\Omega^{(-k)}_{<2\pi f_s>} = (\varpi f_{\text{ss}} - 2\pi k f_s s + \pi f_s) \bmod 2\pi f_s - \pi f_s$. According to assumption 4), the DTFT of $y(n)$ is given in [2] and can be simplified as

$$\begin{aligned} Y(e^{j\omega}) = \frac{1}{T_{\text{ss}}} \sum_{k=0}^{M-1} \sum_{m=0}^{M-1} \Big(e^{j\Omega^{(-k)}_{<2\pi f_s>} T_s \Delta t_m} \\ \times e^{-j\frac{2\pi k}{M} m}\Big) X_a\left(e^{j\Omega^{(-k)}_{<2\pi f_s>}}\right) \end{aligned} \tag{4}$$

where the $\omega = \Omega T_s = 2\pi f / f_s$.

B. Timing Skew Blind Estimation Model

In this section, we will propose a novel TSE model. According to [1], if the analog spectrum $X_a(j\Omega)$ has a specific spectral sparsity property, there are nonoverlapping frequency components ω_p in $Y_m(e^{j\varpi})$, where $\omega_p \in (0, 2\pi)$. The DTFT of the sub-ADC$_0$ output signal at frequency point $\omega_p = (\Omega_p T_{\text{ss}})_{<2\pi>}$ can be expressed as

$$Y_0(e^{j\varpi_p}) = \frac{1}{T_{\text{ss}}} X_a(e^{j\Omega_p}) + \frac{1}{T_{\text{ss}}} \sum_{k=1}^{M-1} X_a\left(e^{j\Omega^{(-k)}_{<2\pi f_s>}}\right). \tag{5}$$

According to the property of the nonoverlapping frequency point, the frequency component of $Y_0(e^{j\varpi})$ at $\varpi_p$only depends on $X_a(j\Omega_p)$, which is the frequency component of $X_a(j\Omega)$ at Ω_p[1]. Therefore, (5) can be rewritten as follows:

$$Y_0(e^{j\varpi}) = X_a(j\Omega_p)/T_{\text{ss}}. \tag{6}$$

Similarly, $\omega^{(q)}_{p\langle 2\pi\rangle} = \Omega^{(q)}_{p\langle 2\pi\rangle} T_s (q = 0, \ldots, M - 1)$ at nonoverlapping frequency point Ω_p. From (4), frequency component $Y(e^{j\varpi})$ at the nonoverlapping frequency point can be expressed as follows:

$$\begin{aligned} &Y\left(e^{j\omega^{(q)}_{p\langle 2\pi\rangle}}\right) \\ &= \frac{1}{T_{\text{ss}}} \sum_{m=0}^{M-1} \left(e^{j\Omega_p T_s \Delta t_m} e^{-j\frac{2\pi(k-q)}{M} m}\right) X_a(e^{j\Omega_p}) \\ &\quad + \frac{1}{T_{\text{ss}}} \sum_{m=0}^{M-1} \sum_{k=0, k\neq q}^{M-1} \left(e^{j\Omega^{(q-k)}_{p\langle 2\pi f_s\rangle} T_s \Delta t_m} e^{-j\frac{2\pi(k-q)}{M} m}\right) X_a\left(e^{j\Omega^{(-k)}_{\langle 2\pi f_s\rangle}}\right) \end{aligned} \tag{7}$$

where q represents the channel that contains the nonoverlapping frequency point. The first term of the sum in (7) represents $Y(e^{j\omega})$ when $k = q$, and the second term of the sum represents $Y(e^{j\omega})$ when $k \neq q$. Based on the property of the nonoverlapping frequency point, when $k \neq q$, the second term of the sum in (7) is 0. Therefore, (7) can be simplified to

$$Y\left(e^{j\omega^{(q)}_{p\langle 2\pi\rangle}}\right) = \frac{1}{T_{\text{ss}}} \sum_{m=0}^{M-1} (e^{j\Omega_p T_s \Delta t_m}) X_a(e^{j\Omega_p}). \tag{8}$$

From (6) and (8), the frequency-domain relationship between $Y_0(e^{j\varpi_p})$ and $Y(e^{j\omega^{(q)}_{p<2\pi>}})$ can be derived as

$$Y\left(e^{j\omega^{(q)}_{p<2\pi>}}\right) = \sum_{m=0}^{M-1} (e^{j\omega_p \Delta t_m}) Y_0(e^{j\varpi_p}). \tag{9}$$

In addition, (9) can be written as

$$\frac{Y(e^{j\omega_p})}{Y_0(e^{j\varpi_p})} = 1 + \sum_{m=0}^{M-1} e^{j\omega_p \Delta t_m}. \tag{10}$$

After FFT, the TIADC system output can be expressed in the form of amplitude and phase angle. The FFTs of system total output $y(n)$ and sub-ADC output $y_m(n)$ can be expressed as follows:

$$\begin{cases} Y(e^{j\omega_p}) = Re^{P(j\omega_p)} \\ Y_m(e^{j\varpi_p}) = R_m e^{P_m(j\varpi_p)} \end{cases} \tag{11}$$

where R and R_m are the amplitudes of $Y(e^{j\omega_p})$ and $Y_m(e^{j\varpi_p})$, respectively, for $m = 0, 1, \ldots, M - 1$, $P(j\omega_p)$ and $P_m(j\varpi_p)$ are the phases of $Y(e^{j\omega_p})$ and $Y_m(e^{j\varpi_p})$, respectively, for $m = 0, 1, \ldots, M - 1$, and $\varpi_p(\omega_p)$ is the nonoverlapping frequency component, which must be estimated. From (11), we obtain

$$\frac{Y(e^{j\omega_p})}{Y_m(e^{j\varpi_p})} = \frac{R}{R_m} e^{(P(j\omega_p) - P_m(j\varpi_p))_{<2\pi>}}. \tag{12}$$

According to the signal decimation theory by M, the decimated signal amplitude is the $1/M$ of original signal amplitude. We obtain the following equation:

$$R/R_m = M. \tag{13}$$

According to (12), the TSE model in [13] can be rewritten as follows:

$$\Delta t_m = \frac{(P_m(j\varpi_p) - P_0(j\varpi_p))_{\langle 2\pi\rangle}}{j\left(\frac{\varpi_p}{M} + 2(k_p - 1)\frac{\pi}{M}\right)_{\langle 2\pi\rangle}} - m, \varpi_p \in (0, 2\pi). \tag{14}$$

We assume that there is timing skew between only one channel of the TIADC system and the first channel. We assume

that the channel is the m_{th} sub-ADC. Then, (10) can be rewritten as follows:

$$\frac{Y(e^{j\omega_p})}{Y_0(e^{j\varpi_p})} = M - 1 + e^{j\omega_p \Delta t_m} (m = 1, \ldots, M-1) \quad (15)$$

$$\omega_p = \left(\frac{\varpi_p}{M} + 2(k_p - 1)\frac{\pi}{M}\right)_{\langle 2\pi \rangle} \quad (16)$$

where k_p and $2k_p - 1$ are the digital-frequency subband indices that are associated with ω_p [1]. As previously described, we assume that sub-ADC$_0$ is the reference channel. Therefore, we define the following expression when there is timing skew only in sub-ADC-m:

$$\Delta P_m = (P_m(j\varpi_p) - P_0(j\varpi_p))_{<2\pi>} \quad (17)$$

where $P_m(j\varpi_p)$ and $P_0(j\varpi_p)$ are the phase expression of $Y_m(e^{j\varpi_p})$ and $Y_0(e^{j\varpi_p})$, respectively. We define

$$\Delta P = (P(j\omega_p) - P_0(j\varpi_p))_{<2\pi>}. \quad (18)$$

Therefore, according to (12), (13), and (15)–(17) we can obtain

$$\Delta t_m = \frac{\ln(Me^{\Delta P_m} - M + 1)}{j\left(\frac{\varpi_p}{M} + 2(k_p - 1)\frac{\pi}{M}\right)_{\langle 2\pi \rangle}} (m = 1, \ldots, M-1). \quad (19)$$

According to assumption 6), ΔP_m tends to 0. Thus, (19) can be reformulated as

$$\Delta t_m = \frac{M\Delta P_m}{j\left(\frac{\varpi_p}{M} + 2(k_p - 1)\frac{\pi}{M}\right)_{\langle 2\pi \rangle}} (m = 1, \ldots, M-1). \quad (20)$$

In the next context, we will introduce a systematic method for estimating the timing skew of the TIADC system using (14) and (20) without considering the gain mismatch or the dc offset mismatch. The specific values of the timing skews of the channels cannot be determined based only on ΔP. ΔP satisfies the following relationship:

$$\Delta P = \Delta P_1 + \Delta P_2 + \ldots + \Delta P_{M-1}. \quad (21)$$

The estimation procedure is as follows.

1) Calculate $\Delta t_1, \Delta t_2, \ldots \Delta t_{M-2}$ using (14).
2) Calculate $\Delta P_1, \Delta P_2, \ldots \Delta P_{M-2}$ using (20).
3) Substitute $\Delta P, \Delta P_1, \Delta P_2, \ldots, \Delta P_{M-2}$ into (21) to obtain ΔP_{M-1}.
4) Obtain Δt_{M-1} using (20).

The estimation procedure of Δt_m is a blind TSE algorithm since only the output spectrum of the TIADC system is required. The proposed SSPR-BLTSE algorithm requires fewer FFT operations for the TIADC system compared with the SS-BLTSE algorithm in [14]. Therefore, this algorithm is of lower computational complexity.

From (14) and (20), the accuracy of the phase spectral estimation determines the estimation accuracy of the timing skews. In this paper, N-point FFT and (N/M)-point FFT are used to compute the sub-ADC spectrum, namely, $Y(e^{j\omega})$ and $Y_m(e^{j\varpi})$, in (4) and (3). TSE accuracy decreases with the increasing of the input spectrum sparsity and the decrease of the data length for FFT. However, performance analysis suggests that FFT-based spectral estimation has multiple limitations, such as "spectral leakage" due to windowing and the picket fence effect, which is caused by the discretization of the spectrum, where the spectral leakage is a continuous function of the frequency and will lead to spreading of the spectrum in different frequency bands [21]. These two factors will affect the estimation accuracy of ω_p and phase spectra. Choosing a proper window function and increasing the data length are effective approaches for reducing the spectral leakage, at the expense of high computational cost. However, this can induce the restriction of online applications. Another possible approach is to use decaying window functions, such as the Hamming window function, to achieve a better tradeoff between computational cost and performance.

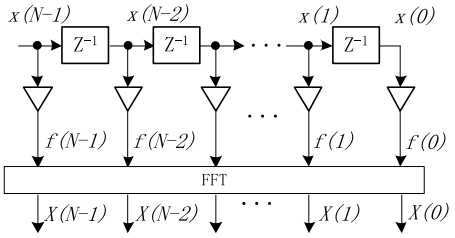


Fig. 2. N-order traditional FFT spectral analysis.

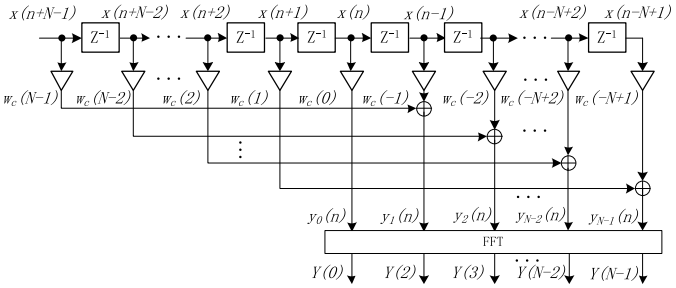


Fig. 3. N-order ApFFT spectral analysis.

III. ApFFT-SSPR-BLTSE Algorithm

In this section, we will apply FFT to the output of the TIADC system after all-phase data preprocessing [7]; this approach is called ApFFT [23]. It is one of the most effective methods for improving the accuracy of the phase spectrum.

A. ApFFT Spectral Estimation Technique

Spectrum leakage is a very serious shortcoming of FFT [24] and severely degrades the spectral analysis. As shown in Fig. 2, FFT with simple windowing is used to improve the spectrum analysis performance. However, compared with the traditional FFT-based spectrum estimation methods, the ApFFT spectral estimation method can inhibit efficiently spectral leakage and has the phase-invariance property. In addition, it can improve substantially the accuracies of the amplitude spectrum and the phase spectrum. Therefore, we can extract precisely the signal phase information without performing any additional corrective actions, even in the case of "nonsynchronous sampling." The structure of ApFFT spectral analysis is illustrated in Fig. 3.

In the next context, a comparison between the analyses of the traditional FFT and ApFFT. We use a single-frequency complex exponential sequence

$$x(n) = e^{j(\omega_0 n + \phi)} = e^{j(nk_0 2\pi/N + \phi)} \quad (22)$$

where ω_0 is the digital frequency, ϕ is the phase, and $k_0 = \text{floor}(\omega_0 N/2\pi)$ is the signal frequency index. The traditional N-point FFT spectrum of $\{x(n)\}$ is

$$\begin{aligned} X(k) &= \frac{1}{N}\sum_{n=0}^{N-1} e^{j\varphi} e^{j2\pi k_0 n/N} e^{-j2\pi kn/N} \\ &= \frac{1}{N} e^{j\varphi} \frac{1-e^{j2\pi(k_0-k)}}{1-e^{j2\pi(k_0-k)/N}} \\ &= \frac{1}{N} \frac{\sin[\pi(k_0-k)]}{\sin(\pi(k_0-k)/N)} e^{j\left[\varphi+\frac{N-1}{N}\pi(k_0-k)\right]} \end{aligned} \quad (23)$$

where $k = 0, 1, \ldots, N-1$. The source of the all-phase data sampling sequence is a $(2N-1)$-length data vector

$$\mathbf{x} = [x(-N+1), x(-N+2), \ldots, x(0), \ldots, x(N-1)]. \quad (24)$$

All N-length shifted data vectors that include $x(0)$ are constructed

$$\begin{aligned} \mathbf{z}_0 &= [x(-N+1), x(-N+2), \ldots, x(0)] \\ \mathbf{z}_1 &= [x(-N+2), x(-N+3), \ldots, x(1)] \\ &\cdots \\ \mathbf{z}_{N-1} &= [x(0), x(1), \ldots, x(N-1)]. \end{aligned} \quad (25)$$

Get another set of N-length data vectors where sample points $x(0)$ is located in the first of each vector through cyclic shifting operation on the N vectors obtained in (25)

$$\begin{aligned} \mathbf{x}_0 &= [x(0), x(-N+1), \ldots, x(-1)] \\ \mathbf{x}_1 &= [x(0), x(1), x(-N+2) \ldots, x(-1)] \\ &\cdots \\ \mathbf{x}_{N-1} &= [x(0), x(1), x(2) \ldots, x(N-1)]. \end{aligned} \quad (26)$$

From (26), a new N-length data vector can be formed by averaging $\mathbf{x}_0, \mathbf{x}_1, \ldots \mathbf{x}_{N-1}$

$$\begin{aligned} x_{\text{ap}} = \frac{1}{N}[Nx(0), (N-1)x(1) + x(-N+1), \ldots, x(N-1) \\ + (N-1)x(-1)]. \end{aligned} \quad (27)$$

The N-length data vector x_{ap} in (27) is the all-phase data vector of a $(2N-1)$-length data vector x. This approach is named the ApFFT spectral estimation method. According to the shift property of FFT, the FFT spectrum of z_i and x_i $(i = 0, \ldots, N-1)$ has the following straightforward relationship:

$$X_i(k) = Z_i(k) e^{j\frac{2\pi}{N}ik} (k = 0, 1, \ldots N-1). \quad (28)$$

Therefore, the FFT spectrum of x_{ap} can be derived as

$$\begin{aligned} X_{\text{ap}}(k) &= \frac{1}{N}\sum_{i=0}^{N-1} X_i(k) \\ &= \frac{1}{N}\sum_{i=0}^{N-1} Z_i(k) e^{j\frac{2\pi}{N}ki} \\ &= \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{n=0}^{N-1} x(n-i) e^{-j\frac{2\pi}{N}kn} e^{-j\frac{2\pi}{N}ki} \end{aligned}$$

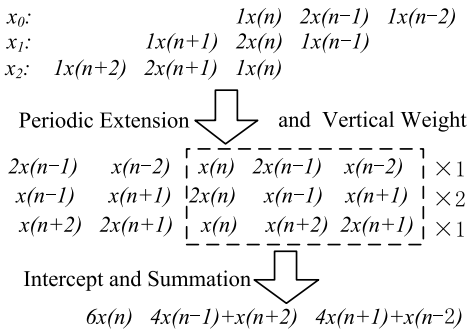


Fig. 4. Block diagram of double-window all-phase data preprocessing.

$$\begin{aligned} &= \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{n=0}^{N-1} e^{j\varphi} e^{j\frac{2\pi}{N}(n-i)k_0} e^{-j\frac{2\pi}{N}kn} e^{-j\frac{2\pi}{N}ki} \\ &= \frac{e^{j\varphi}}{N^2}\sum_{i=0}^{N-1} e^{-j\frac{2\pi(k_0-k)i}{N}} \cdot \sum_{n=0}^{N-1} e^{-j\frac{2\pi(k_0-k)n}{N}} \\ &= \frac{e^{j\varphi}}{N^2} \cdot \frac{1-e^{-j2\pi(k_0-k)}}{1-e^{-j2\pi(k_0-k)/N}} \cdot \frac{1-e^{j2\pi(k_0-k)}}{1-e^{j2\pi(k_0-k)/N}} \\ &= \frac{1}{N^2}\frac{\sin^2[\pi(k_0-k)]}{\sin^2[\pi(k_0-k)/N]} e^{j\varphi}. \end{aligned} \quad (29)$$

By comparing (29) with (23), we determine that the spectral amplitude of $|X_{\text{ap}}(k)|$ that is computed by the ApFFT method is equal to the square of the spectral amplitude of $|X(k)|$ that is computed by the traditional FFT method, which means the attenuation of the sidelobes in $|X_{\text{ap}}(k)|$ is greater than that of those in $|X(k)|$. We conclude that the ApFFT spectral estimation method outperforms the traditional FFT spectral estimation method in terms of spectral leakage.

According to (23) and (29), each phase value of the spectral line in the traditional FFT method is closely related to the frequency deviation value $(k_0 - k)$. However, the phase value of the ApFFT spectrum is ϕ, which is the theoretical phase value of the center sampling point $x(0)$, and ϕ is not related to the frequency deviation value $(k_0 - k)$.

All-phase data preprocessing can be divided into three categories: the no window, single window, and double windows. For different types of convolution windows, double windows outperform no window or a single window in terms of spectral performance. Therefore, as shown in Fig. 4, the double-window data preprocessing method is utilized before FFT in this paper.

B. Proposed ApFFT-SSPR-BLTSE Algorithm

In this section, a new blind TSE algorithm, which is called the ApFFT-SSPR-BLTSE algorithm, will be presented in detail.

Part I (All-Phase Spectrum Estimation): In this part, the ApFFT spectral estimation method will be used to estimate the spectrum of $y(n)$ and $y_m(n)$. The steps, as shown in Fig. 4, are listed as follows.

1) Obtain the $2N-1$-length data vector **x**according to (24) using the $(2N-1)$-length output data of the TIADC.

2) Obtain the shifted data vector $\mathbf{z}_i(i = 0, \ldots, N-1)$ using (24).
3) Use the Hanning window to weight $\mathbf{x}_i(i = 0, \ldots, N-1)$.
4) Calculate the periodic extension in the original position using the weighted sequence and the Hanning window for vertical weight.
5) Obtain a new periodic sequence using the sequence that has been double weighted and the periodic extension for summation in the vertical direction.
6) Truncate the new periodic sequence using the rectangular window and get the double-window all-phase input sequence x_{ap}.
7) Perform N-point FFT on x_{ap} and obtain the all-phase spectrum estimation $Y(e^{j\omega})$.
8) Similarly, estimate all-phase spectrum $Y_m(e^{j\varpi})$ of $y_m(n)(m = 0, 1 \ldots, M-2)$ using the same procedure and use the $(2N/M-1)$-length output data of sub-ADC $(y_m(n))$ to form N/M-length all-phase data vector $\mathbf{x}_{\text{ap}}$.

Part II (Determination of Nonoverlapping Frequency Component ω_p): In this part, we will propose a new method for determining the nonoverlapping frequency point ω_p.

1) Scan the digital frequency from 0 to π, find the maximum-amplitude point $\{\varpi_m : f|Y_0(e^{j\varpi_m})|_{\max}\}$ as the nonoverlapping frequency candidate point.
2) For ϖ_m, the corresponding M amplitude frequency spectrum values are

$$N(k) = \left|Y\left(e^{j\left(\frac{\varpi_m}{M}+2(k-1)\frac{\pi}{M}\right)}\right)\right|, \quad k = 1, \ldots, M. \tag{30}$$

3) Determine the maximum value and the second maximum value of $N(k)$, for $k = 1, \ldots, M$, which are denoted by $N_p(k_p)$ and $N_s(k_s)$, respectively, and calculate the ratio

$$\varepsilon = \frac{N_s(k_s)}{N_p(k_p)}. \tag{31}$$

4) Judge whether ε is smaller than ε_2 and larger than ε_1. We will discuss ε_1 and ε_2 in detail in Experiment 4 in Section IV-B. If

$$\varepsilon_1 \le \varepsilon \le \varepsilon_2. \tag{32}$$

We define $\omega_p = (\varpi/M) + 2(k_p - 1)(\pi/M)$ as the nonoverlapping frequency point and its index k_p can be determined. Otherwise, let $|Y_0(e^{j\varpi})|_{\max} = 0$ and return to step 1) to find the nonoverlapping frequency point. This process is repeated until (32) is satisfied.

Part III: Calculate the Relative Timing Skews Δt_m

1) Calculate the all-phase spectrum $Y(e^{j\omega})$ and the all-phase spectrum $Y_m(e^{j\varpi})(m = 0, 1, \ldots, M-2)$ using the method described in Part I.
2) Find nonoverlapping frequency component ω_p and its index k_p, as described in Part II. Compute the phase spectrum $P(j\omega_p)$ of $Y(e^{j\omega_p})$ and $P_m(j\varpi_p)$ $(m = 0, 1, \ldots, M-2)$ of $Y_m(e^{j\varpi_p})(m = 0, 1, \ldots, M-2)$.
3) Compute the relative timing skews $\Delta t_m(m = 1, 2, \ldots, M-1)$ by (14), (20), and (21), as described in Section II-B.

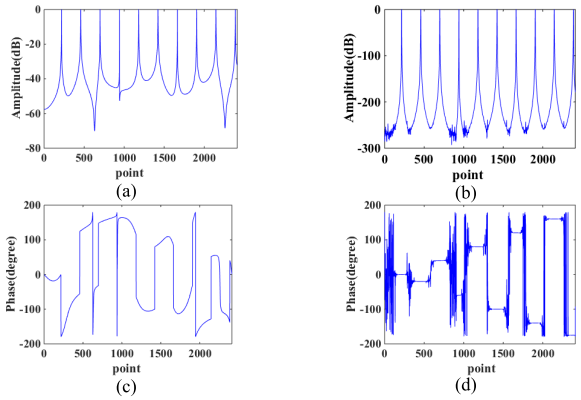


Fig. 5. Comparison of the traditional FFT spectrum and the ApFFT spectrum. (a) Traditional FET amplitude diagram. (b) ApFET amplitude diagram. (c) Traditional FET phase diagram. (d) ApFET phase diagram.

IV. Simulation Results

To evaluate the performances of the ApFFT algorithm and the proposed ApFFT-SSPR-BLTSE algorithm, several experiments are presented in this section. In Section IV-A, the experiment is performed to verify the accuracy of the ApFFT in estimating the phase. In Section IV-B, we simulate a four-channel TIADC system that is assumed to have only timing skews.

A. Phase Estimation Precision of All-Phase FFT for Multifrequency Signal

In this part, the experiment is performed to evaluate the performance of the ApFFT algorithm in calculating the signal phase.

Experiment 1: A multifrequency signal is given as follows:

$$x(t) = \sum_{k=1}^{10} A_k \cos(2\pi f_k + \phi_k) \tag{33}$$

where $f_k = f_1 + (k-1) * 0.05 \times 10^8$, $f_1 = 4.456873$ MHz, $\varphi_k = [0, -20, 40, -60, 80, -100, 120, -140, 160, -175]$, and A_k is an uniformly random distribution sequence, which ranges from 5 to 10. The sampling frequency $f_s = 100$ MHz and the number of FFT points is 2^{12}. The results of the traditional FFT spectral analysis and the all-phase Hanning double window FFT spectral analysis are shown in Fig. 5. The estimated phase values of all frequency points are shown in Table I. According to Fig. 5 and Table I, higher phase estimation accuracy is achieved by the all-phase double-window analysis than by the traditional FFT analysis [25] and the signal phase value can be estimated accurately by 2^{12} point with ApFFT.

B. Performance of the Proposed ApFFT-SSPR-BLTSE Algorithm for the Simulated TIADC System

In this part, we evaluate the performance of the ApFFT-SSPR-BLTSE algorithm in difierent aspects. A four-channel TIADC system is considered, in which the timing skews values are set as $\Delta t_0 = 0, \Delta t_1 = 0.03/f_s$,

TABLE I
RESULTS OF PHASE ESTIMATION

	Phase(Degree)				
Theoretical value	0	-20	40	-60	80
FFT estimated value	0.0300	-55.9392	148.0737	12.1664	116.2014
ApFFT estimated	0.0000	-20.0000	40.0000	-60.0000	80.0000
D-value*(10^{-12})	0.3848	-0.5080	1.3216	1.1511	1.3074
Theoretical value	-100	120	-140	160	-175
FFT estimated value	80.1311	-95.9429	-32.0557	-128.0000	40.8739
ApFFT estimated	-100.0000	120.0000	-140.0000	160.0000	-175.0000
D-value*(10^{-12})	-0.2416	-1.0090	4.1496	-5.4001	-17.2804

* The D-value is the difference between the theoretical value and the ApFFT-estimated value.

TABLE II
SIMULATION RESULTS OF THE ApFFT-SSPR-BLTSE ALGORITHM

		Single-frequency sine wave			Multi-frequency sine wave (L=32)		
Channel index m		1	2	3	1	2	3
True $\Delta m(t_s)$		0.03	0.02	-0.06	0.03	0.02	-0.06
Estimated $\Delta' m(t_s)$	SS-BLTES algorithm in [1]	0.02994	0.01987	-0.05973	0.0305	0.0203	-0.0597
	ApFFT-SSPR-BLTSE algorithm	0.02994	0.01988	-0.05973	0.02998	0.01996	-0.05986
Relative error	SS-BLTES algorithm in [1]	0.20%	0.65%	0.45%	1.6%	1.5%	0.5%
	ApFFT-SSPR-BLTSE algorithm	0.20%	0.60%	0.45%	0.07%	0.20%	0.23%

$\Delta t_2 = 0.02/f_s$, and $\Delta t_3 = -0.06/f_s$, where $f_s = 400$ MHz is the overall sampling frequency of the system.

1) Experiment 2: This experiment evaluates the TSE capability of the proposed ApFFT-SSPR-BLTSE algorithm under noiseless conditions. The input signal is set as follows in two cases.

1) The sinusoidal random signal with random frequency, phase, and amplitude that is given by

$$x_1(t) = A_0 \sin(2\pi f_0 t + \varphi_0) \tag{34}$$

where f_0, A_0, and φ_0 are uniformly distributed random values. The signal frequency f_0 ranges from 1 Hz to 200 MHz, A_0 ranges from 5 to 10 and φ_0 ranges from $-\pi$ to π.

2) The multifrequency random input signals that is given by

$$x_2(t) = \sum_{k=1}^{L} A_k \sin(2\pi f_k t + \varphi_k) \tag{35}$$

where $L = 32$ denotes the number of the input frequencies and f_k, A_k and φ_k are uniformly distributed random sequences and the ranges of the values are the same as for f_0, A_0 and φ_0, respectively.

The total number of samples N for TIADC is defined as 65 536 and the FFT point of each sub-ADC is $N/M = 16\,384$. One hundred independent simulations were conducted for each of the two cases. The mean and variance are estimated, and the simulation results are shown in Table II and Fig. 6. We carried the same experiment as the two cases when $L = 16$ and $L = 64$ in (35). The root-mean-square error (RMSE) is taken as the measure for the four cases that are specified above, which is defined as

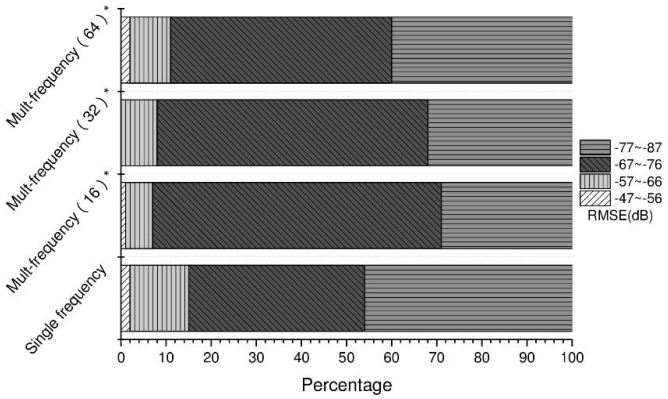


Fig. 6. Statistical results of RMSE. Figures in brackets represent the number of tones of the multifrequency signal.

$$\text{RMSE} = 20\log_{10}\sqrt{\frac{1}{M-1}\sum_{m=1}^{M-1}\left(\Delta_m - \Delta'_m\right)^2}\ (\text{dB}) \tag{36}$$

$$\text{Relative error} = \frac{\left|\Delta_m - \Delta'_m\right|}{\Delta_m} \tag{37}$$

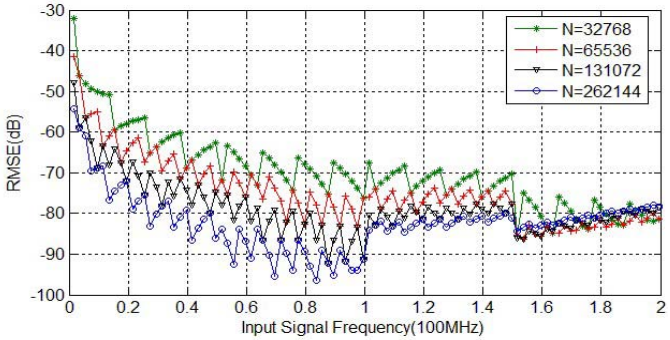


Fig. 7. RMSE versus input frequency.

where Δ_m and Δ'_m denote the true and estimated relative timing skews for the m_{th} sub-ADC, respectively.

According to Table II and Fig. 6, the ApFFT-SSPR-BLTSE algorithm can estimate the channel timing skews effectively with low estimation errors for a single-frequency and multifrequency sine wave. The experimental results can be summarized as follows.

1) The mean relative errors are all below 0.6%, with the extremely low variance.
2) All the estimated results of the RMSE are below −47 dB and most are below −67 dB.
3) The estimation accuracy is better for the multifrequency signal than for the single-frequency signal.

2) Experiment 3: This experiment is carried out to evaluate the impacts of f_i (input frequency) and N (FFT length) on the TSE accuracy of the ApFFT-SSPR-BLTSE algorithm when the input signal is single-frequency sine wave. As in the previous experiment, we use the RMSE as the measure. The other simulation parameters are set to the same values as in Experiment 2. The simulation results are shown in Fig. 7.

According to Fig. 7, the RMSE values of the frequencies are bounded by −60 dB, except for a few of the low-frequency points. That is to say, the estimation accuracy of the proposed algorithm is relatively low for the signal at low frequency because the estimated Δt_m is inversely proportional to the input frequency [1] according to (14) and (20). In addition, RMSE decreases as the number of sampling points N increases. The ApFFT-SSPR-BLTSE algorithm already achieves higher accuracy estimation results. Based on the above analysis, we choose $N = 65\,536$ as a trade-off between the estimation accuracy and the computational complexity.

3) Experiment 4: This experiment evaluates the relationship between parameter ε and the estimation accuracy of the ApFFT-SSPR-BLTSE algorithm using the RMSE measurement, which is defined in (36). As shown in (31), parameter ε is the ratio of the maximum and the second maximum values of the nonoverlapping frequency for the corresponding four points in the amplitude spectrum of the total output of the TIADC system. The four types of input signals that are used in Experiment 2 are also used in this experiment and 200 random experiments are conducted for each type of the input signal. The experiment results are shown in Fig. 8.

From Fig. 8, the RMSE is higher when the parameter ε is smaller, and the accuracy of the estimation method is lower

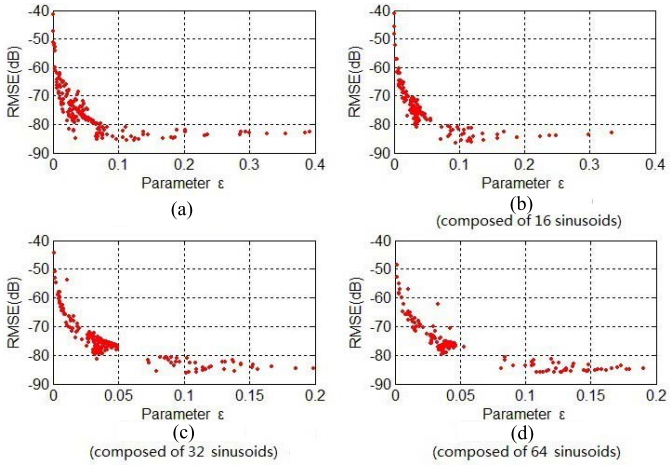


Fig. 8. RMSE versus ε in different signals. (a) Single frequency. (b) Multifrequency (composed of 16 sinusoids). (c) Multifrequency (composed of 32 sinusoids). (d) Multifrequency (composed of 64 sinusoids).

in this situation. Furthermore, a small value of parameter ε corresponds to a low-frequency signal according to an analysis of the experimental data. Therefore, the value of parameter ε should not be too small when selecting the nonoverlapping frequency point. However, a larger value of parameter ε could be the result of signal aliasing. We conclude that each channel timing skew can be estimated accurately when parameter ε satisfies (32). From the experimental results of Fig. 8, we can obtain the empirical values of ε_1 and ε_2 for 0.006 and 0.2, respectively. The RMSE will reach below −60 dB. In addition, we can further improve the estimation accuracy of the algorithm by increasing the value of parameter ε_1 appropriately. Poor estimation accuracy can be efiectively avoided by using a low-frequency signal.

4) Experiment 5: This experiment aims at evaluating the impact of the variation of the timing skews. In this paper, timing skews Δt_1 and Δt_2 are evaluated by using (14) directly and timing skew Δt_3 is not evaluated directly. According to the difierent estimation methods of the timing skews in each channel, the parameter settings in this experiment can be divided into the following two groups.

1) $\Delta t_2 = 0.02$, $\Delta t_3 = -0.06$, and Δt_1 is set from $-0.10\ T_s$ to $-0.10\ T_s$ with a step size of $0.002\ T_s$.
2) $\Delta t_1 = 0.03$, $\Delta t_2 = 0.02$, and Δt_3 is set from $-0.10\ T_s$ to $-0.10\ T_s$ with a step size of $0.002\ T_s$.

The other simulation parameters are set to the same values as in Experiment 2, with the input signal generated by (35) and $L = 16$. One hundred random simulations are conducted for each value of the timing skew and the mean relative error and variance of the estimation results are calculated. The overall results are shown in Fig. 9.

According to (37), the relative error of the timing skew near the zero point is large, as shown in Fig. 9(a), because the absolute value of the timing skew is small at these points. The variance is very small at these points, as shown in Fig. 9(b). In Fig. 9(b), the total variance values are below −130 dB and almost constant with the change of timing skews. Thus, we conclude that the timing skew does not affect the estimation accuracy of the ApFFT-SSPR-BLTSE algorithm.

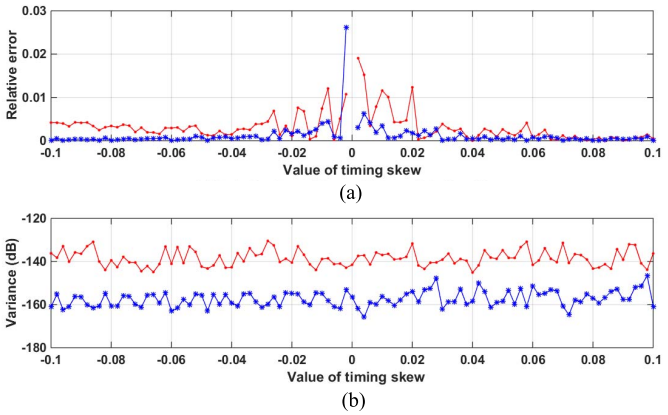


Fig. 9. Estimation results with the variance. (a) Relative error versus value of timing skew. (b) Variance versus value of timing skew.

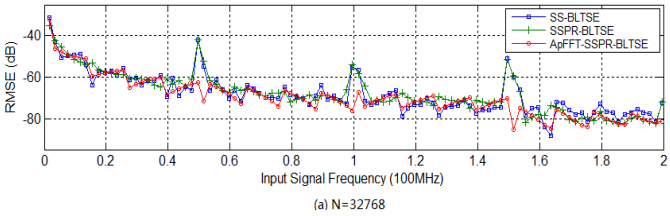


Fig. 10. Relationship between RMSE and input signal frequency.

5) Experiment 6: This experiment is carried out to compare the SS-BLTSE, SSPR-BLTSE, and ApFFT-SSPR-BLTSE algorithms, and includes the following two subexperiments.

a) Relationship between RMSE and input signal frequency: The experimental method that is used in Experiment 3 is also used in this experiment, and the results are shown in Fig. 10. From Fig. 10, we conclude that using the ApFFT method can improve the stability and the estimation accuracy of the estimation algorithm when the input signal frequency is 50, 100, or 150 MHz. That is to say, the spectral leakage can be suppressed by ApFFT at these frequency points.

b) Relationship between RMSE and parameter ε: The experimental method that is used in Experiment 4 is also used in this experiment, and the results are shown in Fig. 11. From Fig. 11, the following conclusions are obtained.

1) The estimation accuracies of the three algorithms are similar when the input signal is a single-frequency signal. The performance in terms of RMSE is not substantially improved compared with the SS-BLTSE algorithm and ApFFT-SSPR-BLTSE algorithm. However, the proposed ApFFT-SSPR-BLTSE algorithm has lower computational complexity than other algorithms.

2) The estimation accuracy of the ApFFT-SSPR-BLTSE algorithm is superior to those of other algorithms when the input signal is a multifrequency signal, and the ApFFT-SSPR-BLTSE algorithm has lower computational complexity than others.

No matter how much number of tones of the input signal, the RMSE are all below -60 dB by using the algorithm of the ApFFT-SSPR-BLTSE.

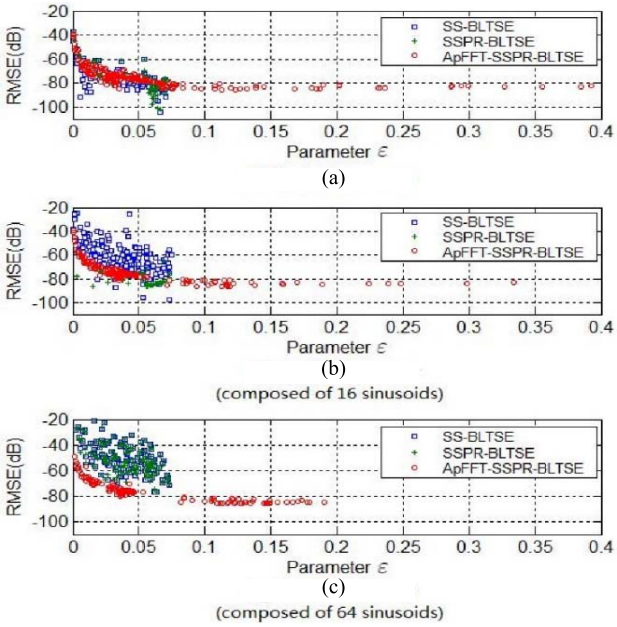


Fig. 11. Relationship between RMSE and parameter ε. (a) Single frequency. (b) Multifrequency (composed of 16 sinusoids). (c) Multifrequency (composed of 64 sinusoids).

To compare the ApFFT-SSPR-BLTSE algorithm and the SS-BLTSE algorithm in [1] in terms of estimation effectiveness, we use the estimated timing skew parameters that are listed in Table II to compensate the timing mismatches. In this experiment, the frequencies of the two-tone input signal are 61 and 110 MHz and the sampling frequency of the four-channel TIADC system is $f_s = 400$ MHz. For visualization purposes, the compensated output spectra of the original TIADC system and the TIADC system that is assumed to have only timing skews are plotted in Fig. 12. The output spectrum spurs without compensation are shown in Fig. 12(a). According to Fig. 12(b), the spectrum spurs, which are due to the timing skews of the TIADC system, have been effectively compensated using the proposed ApFFT-SSPR-BLTSE algorithm. From Fig. 12(b) and (c), the proposed ApFFT-SSPR-BLTSE algorithm has better estimation effectiveness than the SS-BLTSE algorithm in [1]. In addition, the SFDR after the calibration can be enhanced from 20.18 to 80.23 dB for the ApFFT-SSPR-BLTSE algorithm and from 20.18 to 75.16 dB for the SS-BLTSE algorithm.

V. Hardware Implementation

The overall architecture of the proposed ApFFT-SSPR-BLTSE algorithm is shown in Fig. 13. It has three main blocks: 1) ApFFT unit; 2) estimation of nonoverlapping frequencies (ENF); and 3) timing skew computation unit. In addition, it has memory and control units. The mismatched TIADC output of the total system $y(n)$ and the output of m_{th} sub-ADC $y_m(n)$ are stored in the memory units. We choose a sine signal as the input of the four-channel TIADC system with $N = 65\,536$ sample point in this design. In the ApFFT unit, the IPs of FFT, which are used to process data in parallel for

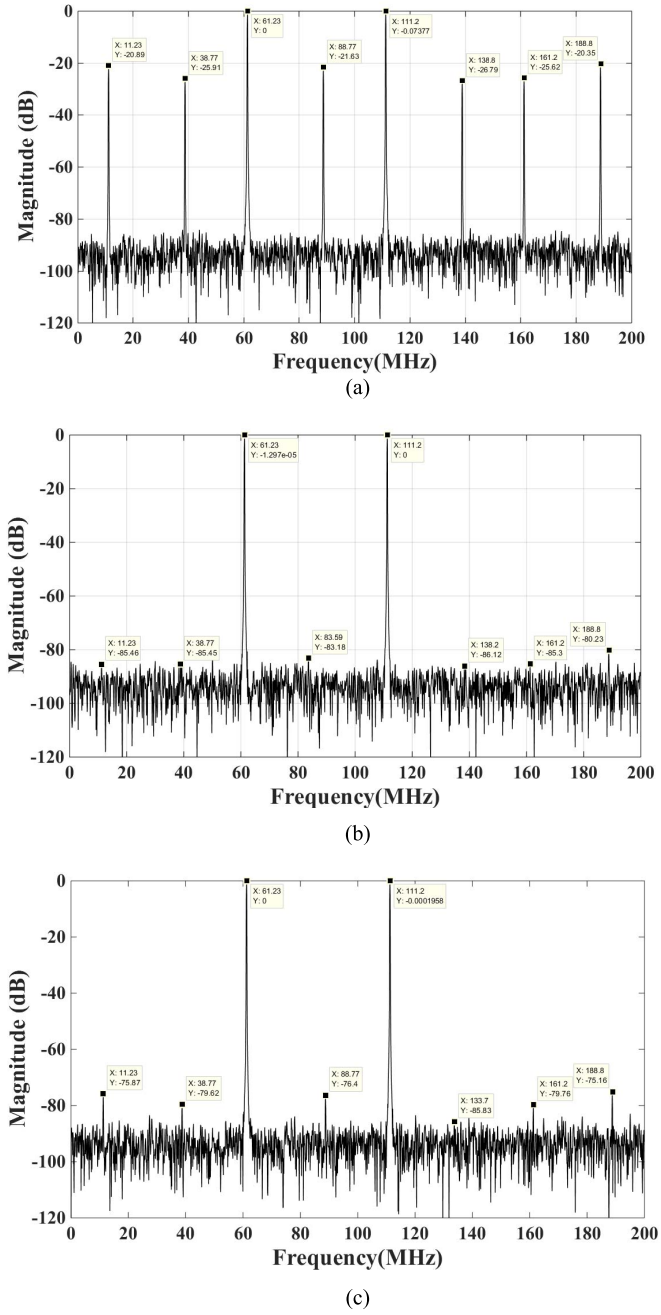


Fig. 12. (a) Original spectrum of input signal. (b) Compensated output spectrum (TSE by the ApFFT-SSPR-BLTSE algorithm). (c) Compensated output spectrum (TSE by the SS-BLTSE algorithm).

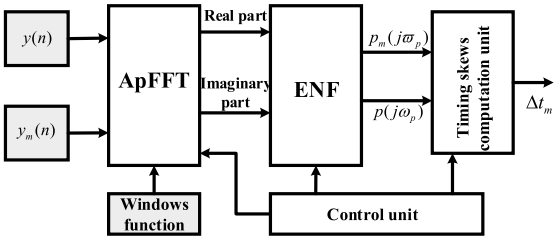


Fig. 13. Architecture of the ApFFT-SSPR-BLTSE algorithm.

output $y(n)$ and $y_m(n)$, receives the discrete output signals of the TIADC system and computes the ApFFTs $Y(e^{j\omega})$ and $Y_m(e^{j\varpi})$. The nonzero results of the ApFFT unit are stored in

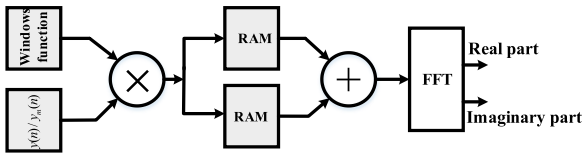


Fig. 14. Structure of ApFFT.

memory units. The ENF unit receives the nonzero results from the memory units, finds the frequency component of maximum amplitude point ω_p and calculates the phase spectrum $P(j\omega_p)$ and $P_m(j\varpi_p)$. The timing skew computation unit calculates the relative timing skews $\Delta t_m (m = 1, 2, \ldots, M-1)$ using (14), (20), and (21). The control unit generates the address values for performing read and write operations on the memory unit, and sends the necessary control signals to the computing units to initiate their operations.

A. Proposed Structure of ApFFT

The structure of ApFFT is shown in Fig. 14 for $N = 65\,536$, $M = 4$, and sampling frequency of TIADC system $f_s = 400$ MHz. It is composed of two IPs of FFT with different data sizes, multipliers, and adders. In this design, we set the number of sample points for subchannels as N/M and the number of total output sample points as N. Therefore, N-point FFT and N/M-point FFT are used to calculate all-phase spectra of the output data. The output of the TIADC system $y(n)$ and the output of sub-ADC $y_m(n)$ are stored in RAMs. The main difference between the traditional FFT and the ApFFT is that the data are preprocessed in different ways. In this unit, the double-window data preprocessing method is implemented before FFT. Therefore, to simplify processing, we can store the calculated data of the window functions into RAMs before the calculation starts. Since the design uses two different signal lengths, two different lengths of window functions are also required. During the N cycles of each channel for the computation of ApFFT, $y_m(n)$, $y(n)$ and corresponding window functions can be retrieved separately from the RAMs in serial order and fed to the multiplier unit. For each channel, the results of multiplication are divided into two equal parts and can be stored in two shared RAMs of the same size to complete the addition operation.

The operations of subchannels and total output are performed at the same time to improve the parallelism. After the preprocessing of the data, the results are sent to the FFT units and all-phase spectra $Y_m(e^{j\varpi})$ and $Y(e^{j\omega})$ are obtained. The ApFFT unit has accomplished the work of Part I: all-phase spectral estimation, as described in Section III.

B. Proposed Structure of the ENF Unit

The ENF unit consists of a judging unit, a square sum unit, an Arctan unit, a computation unit, and a comparison unit. Since the input signal of the TIADC system is sparse, the results of only a few points after the FFT are nonzero. In this unit, we set a threshold for determining the nonzero value in the judging unit and store separately the imaginary

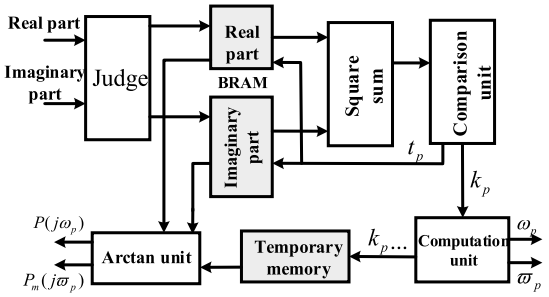


Fig. 15. Internal structure and data flow of the ENF unit.

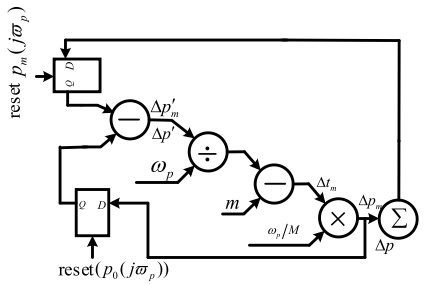


Fig. 16. Data flow of the timing skew computation unit.

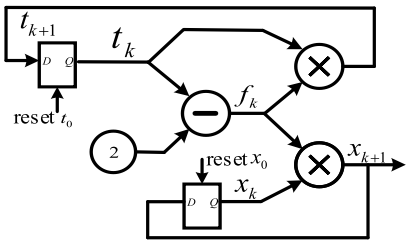


Fig. 17. Internal structure of the Goldschmit algorithm.

part and the real part in the RAMs. The square of the amplitude can be calculated in the square sum unit and the nonoverlapping frequency point can be found in the comparison unit. The comparison unit generates the temporary ordering t_p of the maximum value and the real location k_p. The comparison unit can arrange all the phase values in descending order. The input data of the comparison unit are positive and serial and we can scan sequentially the input data. In this design, we only select the two maximum values for calculating the ratio in (31) for every cycle until (32) is satisfied. Otherwise, the maximum value is set to 0 in the next cycle. Finally, the temporary ordering t_p can be used as an address to obtain the maximum value from the RAMs of the real part and the imaginary part. In the computation unit, ϖ_p can be calculated according to $\varpi_p = \pi k_p/N/2$ and ω_p can also be calculated using (16). Therefore, the computation unit consists of a division unit, multiplication unit, and an addition (subtraction) unit. These units can be reused in the timing skew computation unit. Then, the IP of the Arctan unit calculates phase spectrum $P(j\omega_p)$ and $P_m(j\varpi_p)$ for $m = 0, 1, 2$.

Fig. 15 shows the internal structure and the data flow of the ENF unit. In this design, we first determine the position of the nonoverlapping frequency point according to the first channel (the reference channel) of the TIADC system. Then, we calculate the phase spectra $P_m(j\varpi_p)$ of the other subchannels and $P(j\omega_p)$ of the total output at frequency point ω_p. The results of other subchannels and the total output through the ApFFT unit are stored into temporary memory and can be used to obtain the phase spectrum. In this design, temporary memory is implemented in some registers.

C. Timing Skew Computation Unit

According to (14), (20), and (21), we obtain the timing skews of each channel in the TIADC system. The internal structure and data flow of the timing skew computation unit are shown in Fig. 16. In this design, we optimize the architecture by inserting pipeline registers for different operations to improve the performance of this design. Phase spectra $P(j\omega_p)$ and $P_m(j\varpi_p)$ can be obtained by the ENF unit and the intermediate value by the subtraction operation. The intermediate value and ω_p are inputs of the division unit. Many division algorithms and implementations have been published in [26] and [27]. In this paper, to invert ω_p in the division unit, we have developed a reciprocal operator that is based on the Goldschmit method for calculating the inverse of ω_p. The internal structure of the Goldschmit method is shown in Fig. 17. Parallel computing can double the computational speed of the division operation and the iterative formula is defined in (38), which can be calculated from x_i using only multiplication and subtraction. First, we standardize parameters C and D. Then, we initialize $x_0 = C = 1$ and $t_0 = D(0.5 \le D < 1)$. Finally, the cycle is repeated until the required accuracy is obtained by the iterative operation. The Goldschmidt algorithm is normally implemented by using an initial approximation (see value). Then, k iterations, where k depends on the accuracy of the initial approximation and the required final precision, are performed.

The result of the division operation and m are inputs of the subtraction unit and are used to calculate Δt_m by (14). Then, the multiplication unit receives Δt_m and ω_p/M and generates Δp_m using (20). After obtaining $\Delta p, \Delta p_1, \Delta p_2, \ldots, \Delta p_{M-2}$, we calculate Δp_{M-1} using (21). In this process, accumulators can be utilized and the subtraction unit can be reused. Finally, Δt_{M-1} is calculated according to (20) and the multiplication unit is reused

$$\begin{cases} f_k = 2 - t_k \\ x_{k1} = x_k \times f_k \\ t_{k1} = t_k \times f_k. \end{cases} \tag{38}$$

VI. Results of FPGA Implementation

In this section, we present the implementation and performance evaluation of the proposed design, and the results in the case of $N = 65\,536$, $M = 4$, and $f_s = 400$ MHz.

A. FPGA Implementation Approach

To implement the ApFFT-SSPR-BLTSE algorithm on the hardware, an estimation algorithm is designed and optimized in terms of the fixed points of signals using MATLAB. In this

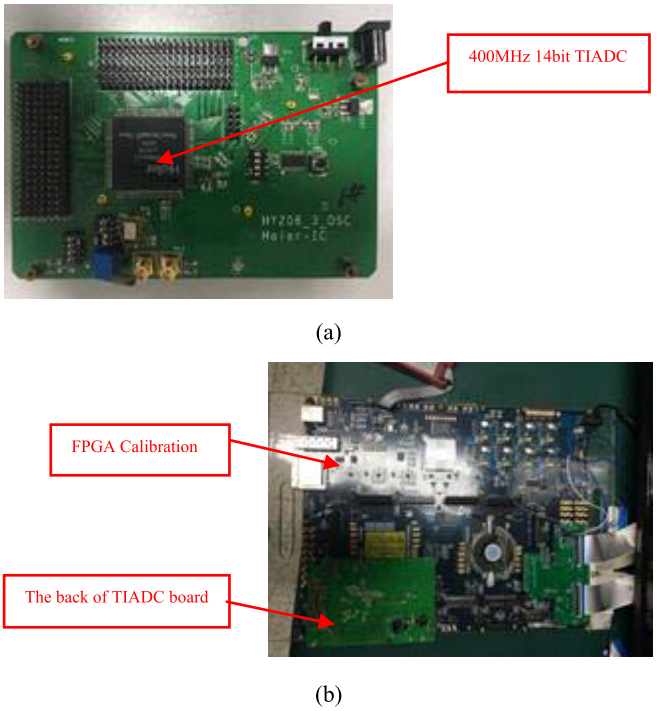


Fig. 18. (a) Four-channel 14-bit 400-MHz TIADC system. (b) FPGA calibration platform.

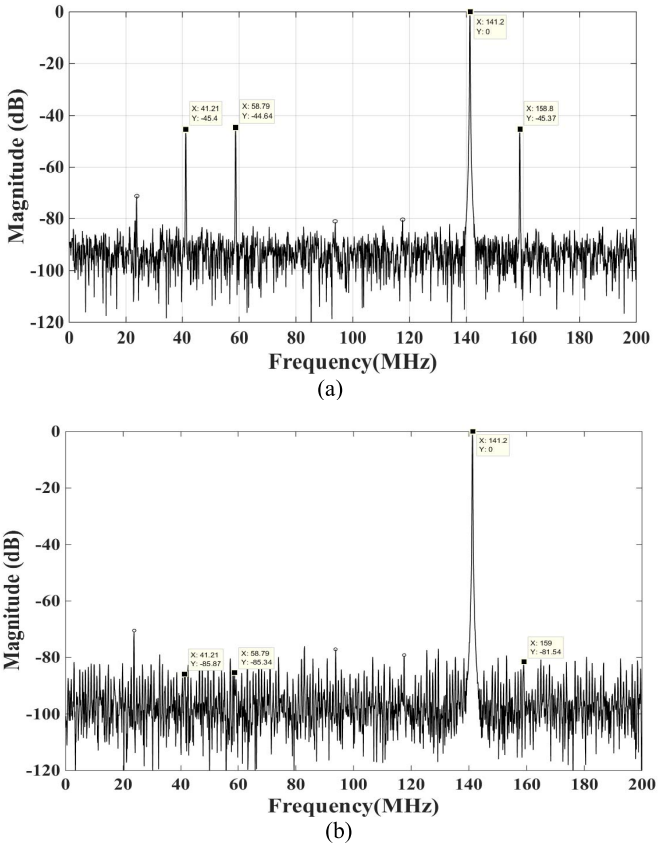


Fig. 19. (a) Measured output spectrum before calibration. (b) Measured output spectrum after calibration.

design, we use 25-bit data precision with 20 bits for the fractional part. We have transformed the abstract representation of our system-level design into a hardware description language code for register transfer level-level representation. The hardware operators that are included in the design have been optimized to reduce their critical paths as much as possible. All operators (addition, accumulation, multiplication and division) have been designed in a pipeline fashion to boost the circuit performances. The model of the proposed architecture is designed using a fixed-point data representation. The precision (number of bits) and the scale (binary point) of the fixed-point data are defined as parameters. To validate the proposed hardware implementation, we have developed a MATLAB code of the ApFFT-SSPR-BLTSE algorithm.

TABLE III
HARDWARE IMPLEMENTATION RESULTS

Device	6vlx550tff1759
Max Frequency	116.82MHz
BRAM	36/632 (45%)
Logic registers	21418/687360(3%)
Total registers	21418
DSP slices	118/864(13%)
LUTs	19495/343680 (5%)

B. Implementation Results

Considering the requirements of high performance and low power consumption, the proposed architecture of the ApFFT-SSPR-BLTSE algorithm is synthesized and placed on a Xilinx Virtex-6 vlx550tff1759 FPGA device. Xilinx Virtex-6 FPGA represents advanced integrated-chip technology and adopts 40-nm FPGA technology. Power consumption and area reduction are not addressed in this paper since the main objective is to provide higher clock frequency. The architecture logic synthesis was performed using the Xilinx ISE 14.7 tool.

The TIADC board is shown in Fig. 18(a). It was improved on the basis of [12] by our team. The TIADC in Fig. 18(a) is plugged into the FPGA board in Fig. 18(b). [The back of the TIADC board is shown in the left of Fig. 18(b).] The measured data from the 400 MHz, four-channel, 14-bit TIADC board are fed to the FPGA board. The data are calibrated through the FPGA board and sent to a PC through the USB interface for analysis of the spectrum by MATLAB. The spectral plots, which aim at comparing the output spectra before and after calibration using the proposed calibration method, are shown in Fig. 19(a) and (b) with the input signal frequency of 141.2 MHz. Comparing Fig. 19(a) and (b), the mismatch spurs were suppressed to better than -81.54 dB. After calibration, the mismatch distortions were no longer visible above the noise floor. Spurs that are marked as "o" are the harmonics that are caused by the nonlinearity of the sub-ADCs, which cannot be calibrated by the proposed algorithm. The validation results demonstrate the high performance of the algorithm. As shown in Table III, the design can operate on the FPGA at a clock frequency of 116.82 MHz and consumes only a few percent of the hardware resources in the FPGA chip.

TABLE IV
COMPARISON OF COMPUTATINAL COMPLEXITY

Computational complexity (number of operations)	SS-BLTES algorithm in [1]	Proposed algorithm
N-point FFT	1	1
(N/M)-point FFT	M	M-1
multiplication	M+1	M
division	M+1	M
subtraction	M+1	M

C. Analysis of Computational Complexity

The computational complexity of the proposed architecture is evaluated in this part. As described in Section II, an M-channel TIADC system generates a total of N discrete data samples. Compared with the SS-BLTES algorithm in [1], the proposed ApFFT-SSPR-BLTSE algorithm decreases the computational complexity as shown in Table IV. The computational complexities of the SS-BLTES algorithm in [1] and the proposed ApFFT-SSPR-BLTSE algorithm mainly originate from three parts.

1) An N-point FFT for obtaining $Y(e^{j\omega})$ has a computational complexity of $N \log N$.
2) For each sub-ADC output, the calculation of the SS-BLTES algorithm in [1] requires M (N/M)-point FFT operations and that of the proposed ApFFT-SSPR-BLTSE algorithm requires $M-1$ (N/M)-point FFT operations. Reducing the number of FFT operations will reduce dramatically the calculation burden of the algorithm.
3) The computational complexity of determining ω_p and calculating the relative timing skews $\Delta t_m (m = 1, 2, \ldots, M-1)$ using (14), (20), and (21) is slightly increased in the proposed ApFFT-SSPR-BLTSE algorithm compared with the SS-BLTES algorithm in [1].

In conclusion, the proposed ApFFT-SSPR-BLTSE algorithm is of lower computational complexity because $N \gg M$.

In addition, we compare the computational complexity of the proposed approach with that of the approach in [14]. As mentioned in [14], the correction involves a discrete Fourier transform (DFT) calculation and an inverse DFT calculation, which require $N \log_2 N$ operations each per ADC. This results in a total of $M^2 N + 2MN \log_2 N$ operations for the correction of each data batch. Here, one operation is defined as one multiplication and one summation. However, in this paper, an N-point FFT for obtaining $Y(e^{j\omega})$ has a computational complexity of $N \log_2 N$, and $M-1 (N/M)$-point FFTs are required. In conclusion, the computational complexity of the proposed ApFFT-SSPR-BLTSE algorithm is approximately $o(2N \log_2 N)$. Therefore, the ApFFT-SSPR-BLTSE algorithm can be considered a low-computational-complexity estimation method.

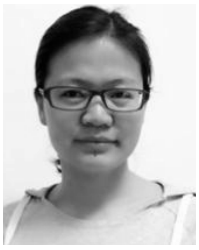
VII. CONCLUSION

In this paper, the ApFFT-SSPR-BLTSE algorithm has been proposed. The channel timing skew parameters can be determined effectively by the phase relationship at the nonoverlapping frequency points. The ApFFT-SSPR-BLTSE algorithm can reduce the number of FFT operations and has low hardware complexity. In addition, the ApFFT-SSPR-BLTSE algorithm uses the ApFFT spectral estimation technique, which can estimate the signal phase accurately. Simulations show that the mean estimation error of the ApFFT-SSPR-BLTSE algorithm is below 0.6%, with extremely low variance. Finally, the proposed architecture is implemented and validated on the Xilinx Virtex-6 vlx550tff1759 FPGA with only a few percent of the hardware resources of the FPGA chip. It is applied to the four-channel 400-MHz 14-bit TIADC real system, and the mismatch spurs are suppressed to better than -81.54 dB. Compared with the SS-BLTSE algorithm, the proposed algorithm has significant advantages in terms of both estimation accuracy and computational complexity. We are confident in concluding that our proposed ApFFT-SSPR-BLTSE algorithm is a good candidate for scenarios in which real-time high-accuracy TSE is required.

REFERENCES

[1] Y. X. Zou and X. J. Xu, "Blind timing skew estimation using source spectrum sparsity in time-interleaved ADCs," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 6, pp. 2401–2412, Sep. 2012.
[2] C. Vogel, "The impact of combined channel mismatch effects in time-interleaved ADCs," *IEEE Trans. Instrum. Meas.*, vol. 56, no. 1, pp. 415–427, Feb. 2005.
[3] A. Petraglia and S. K. Mitra, "Analysis of mismatch effects among A/D converters in a time-interleaved waveform digitizer," *IEEE Trans. Instrum. Meas.*, vol. 40, no. 5, pp. 831–835, Oct. 1991.
[4] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara, and K. Kobayashi, "Explicit analysis of channel mismatch effects in time-interleaved ADC systems," *IEEE Trans. Circuits Syst. I, Fundam. Theory Appl.*, vol. 48, no. 3, pp. 261–271, Mar. 2001.
[5] M. Seo, M. J. W. Rodwell, and U. Madhow, "Comprehensive digital correction of mismatch errors for a 400-Msamples/s 80-dB SFDR time-interleaved analog-to-digital converter," *IEEE Trans. Microw. Theory Techn.*, vol. 53, no. 3, pp. 1072–1082, Mar. 2005.
[6] M. Seo, M. J. W. Rodwell, and U. Madhow, "Blind correction of gain and timing mismatches for a two-channel time-interleaved analog-to-digital converter," in *Proc. 39th Asilomar Conf. Signals, Syst. Comput.*, Oct. 2005, pp. 1121–1124.
[7] S. J. Liu, X. J. Xu, and Y. X. Zou, "Blind timing skew estimation based on spectra sparsity and all phase FFT for time-interleaved ADCs," in *Proc. IEEE Int. Conf. Digit. Signal Process. (DSP)*, Jul. 2015, pp. 926–930.
[8] C. Vogel, D. Draxelmayr, and G. Kubin, "Spectral shaping of timing mismatches in time-interleaved analog-to-digital converters," in *Proc. Int. Symp. Circuits Syst.*, May 2005, pp. 1394-1397.
[9] Y. C. Lim, Y. X. Zou, J. W. Lee, and S. C. Chan, "Time-interleaved analog-to-digital-converter compensation using multichannel filters," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 10, pp. 2234–2247, Oct. 2009.
[10] Y. X. Zou, S. L. Zhang, Y. C. Lim, and X. Chen, "Timing mismatch compensation in time-interleaved ADCs based on multichannel Lagrange polynomial interpolation," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 4, pp. 1123–1131, Apr. 2011.
[11] H. Johansson and P. Löwenborg, "Reconstruction of nonuniformly sampled bandlimited signals by means of digital fractional delay filters," *IEEE Trans. Signal Process.*, vol. 50, no. 11, pp. 2757–2767, Nov. 2002.
[12] S. J. Liu, P. P. Qi, J. S. Wang, M. H. Zhang, and W. S. Jiang, "Adaptive calibration of channel mismatches in time-interleaved ADCs based on equivalent signal recombination," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 2, pp. 277–286, Feb. 2014.
[13] W. C. Black and D. A. Hodges, "Time interleaved converter array," *IEEE J. Solid-State Circuits*, vol. SSC-15, no. 6, pp. 1022–1029, Dec. 1980.

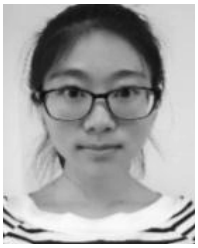
[14] J. Elbornsson, F. Gustafsson, and J.-E. Eklund, "Blind equalization of time errors in a time-interleaved ADC system," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1413–1424, Apr. 2005.
[15] Y.-C. Jenq, "Improving timing offset estimation by alias sampling," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 7, pp. 1376–1378, Jul. 2008.
[16] S. M. Jamal, D. Fu, M. P. Singh, P. J. Hurst, and S. H. Lewis, "Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 1, pp. 130–139, Jan. 2004.
[17] C. Vogel, "A frequency domain method for blind identification of timing mismatches in time-interleaved ADCs," in *Proc. 24th Norchip Conf.*, vol. 296. Nov. 2006, pp. 45–48.
[18] Y. X. Zou, B. Li, and X. Chen, "An efficient blind timing skews estimation for time-interleaved analog-to-digital converters," in *Proc. 17th Int. Conf. Digit. Signal Process.*, Corfu, Greece, Jul. 2011, pp. 1–4.
[19] S. Huang and B. C. Levy, "Blind calibration of timing offsets for four-channel time-interleaved ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 4, pp. 863–876, Apr. 2007.
[20] S. Liu, J. Cui, H. Ma, and Y. Zou, "Blind timing error estimation based on the phasic relationship between nonoverlapping frequency points in time-interleaved ADCS," in *Proc. 12th IEEE ICSICT*, Oct. 2014, pp. 1–3.
[21] P. Stoica and R. L. Moses, *Spectral Analysis of Signals*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2005.
[22] R. S. Prendergast, B. C. Levy, and P. J. Hurst, "Reconstruction of band-limited periodic nonuniformly sampled signals through multirate filter banks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 8, pp. 1612–1622, Aug. 2004.
[23] Z. H. Wang, Z. X. Hou, and F. Su, "All phase FFT spectrum analysis," *J. China Inst. Commun.*, vol. 24, no. 11A, pp. 16–20, 2003.
[24] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing: Principles Algorithms and Applications*, 3rd ed. New York, NY, USA: Macmillan, 1996.
[25] X. Huang, Z. Wang, and G. Hou, "New method of estimation of phase, amplitude, and frequency based on all phase FFT spectrum analysis," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst.*, Nov. 2007, pp. 284–287.
[26] D. Piso, J.-A. Piñeiro, and J. D. Bruguera, "Analysis of the impact of different methods for division/square root computation in the performance of a superscalar microprocessor," *J. Syst. Arch.*, vol. 49, nos. 12–15, pp. 543–555, 2003.
[27] J.-A. Pineiro and J. D. Bruguera, "High-speed double-precision computation of reciprocal, division, square root, and inverse square root," *IEEE Trans. Comput.*, vol. 51, no. 12, pp. 1377–1388, Dec. 2002.
[28] A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Discrete-Time Signal Processing*, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall, 1998.



Sujuan Liu (M'17) received the B.S. degree in electronic engineering from Shandong University, Jinan, China, in 2001, and the Ph.D. degree in microelectronics and solid-state electronics from the Beijing University of Technology, Beijing, China, in 2006.

Since 2006, she has been with the College of Microelectronics, Beijing University of Technology, where she is currently an Associate Professor. She is involved in time-interleaved analog-to-information converter research project. Her current research interests include design and implementation of analog-to-digital converters, mixed signal systems, and digital signal processing.

Dr. Liu has served as the Evaluation/Peer Review Expert for the National Natural Science Foundation of China since 2011 and also as the Paper Reviewer for several IEEE journals and international conferences.



Ning Lyu received the B.S. degree in microelectronics from the Xi'an University of Posts and Telecommunications, Xi'an, China, in 2015. She is currently pursuing the M.S. degree with the College of Microelectronics, Beijing University of Technology, Beijing, China.

Her current research interests include design and implementation recovery algorithms for analog-to-digital converters based on compressive sensing, mixed-signal circuits, and system design.

Jiashuai Cui received the B.S. degree in applied physics from Hangzhou Dianzi University, Hangzhou, China, in 2013, and the M.S. degree from the College of Electronic Information and Control Engineering, Beijing University of Technology, Beijing, China, in 2016.

His current research interests include high-speed data conversion, mixed-signal circuits, and system design.

Yuexian Zou (S'96–M'00–SM'08) received the M.Sc. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 1991, and the Ph.D. degree from the University of Hong Kong, Hong Kong, in 2000.

She is currently a Full Professor of Peking University, Beijing, China, and the Director of the Advanced Data and Signal Processing Lab, Peking University Shenzhen Graduate School, Shenzhen, China. She is involved in the high-speed and high-resolution analog-to-digital converter project. Her current research interests include signal and information processing, array signal processing, and pattern and machine learning.