

An Amplitude Resolution Improvement of an RF-DAC Employing Pulswidth Modulation

Min Park, *Member, IEEE*, Michael H. Perrott, *Member, IEEE*, and Robert Bogdan Staszewski, *Fellow, IEEE*

Abstract—We propose a time-domain technique that significantly improves resolution of an RF-DAC. As an alternative to resorting to various resolution improvement attempts in the amplitude domain or through quantization noise shaping, *pulswidth modulation* (PWM) of a single unit switching device is employed with fine timing accuracy easily afforded by advanced CMOS technology. The PWM is categorized into *centered PWM* and *noncentered PWM* depending on the relative pulse position, and their performance and implementation methods are compared. The technique is examined in the context of a commercial EDGE polar transmitter realized in 65 nm CMOS, which employs an amplitude modulator with basic 10-bit amplitude resolution limited by the RF-DAC switching device mismatches. The proposed architectures with centered PWM and noncentered PWM achieve the worst case resolution improvement of 2.2 bits and 2.5 bits, respectively, assuming 20-ps worst case time granularity of the PWM signal controls.

Index Terms—Amplitude modulation (AM), Enhanced data rate for GSM evolution (EDGE), envelope modulation, polar transmitter, pulswidth modulation (PWM), RF-DAC.

I. INTRODUCTION

DIGITAL functionality is crucial nowadays for most wireless applications. For low-cost and low-power wireless devices, *system-on-chip* (SoC) integration of RF circuits with digital circuits would be ideal. Some recent publications on RF transmitters have shown that employing an RF-DAC can make RF SoC implementation easier [1]–[4]. Among them, the commercial single-chip GSM/EDGE transceiver in [1], [2] is unique in that it uses a simple array of unit-weighted transistor switches to control the output RF amplitude, which operates as near-class-E power amplifier, instead of using a traditional current-source based DAC structure.

Fig. 1 illustrates the polar transmitter introduced in [1], [2]. The I/Q baseband data is converted into digital amplitude modulation (AM) and phase/frequency modulation (PM/FM) signals. The frequency signal is fed into the DCO-based N_F -bit *digital-to-frequency converter* (DFC), which generates

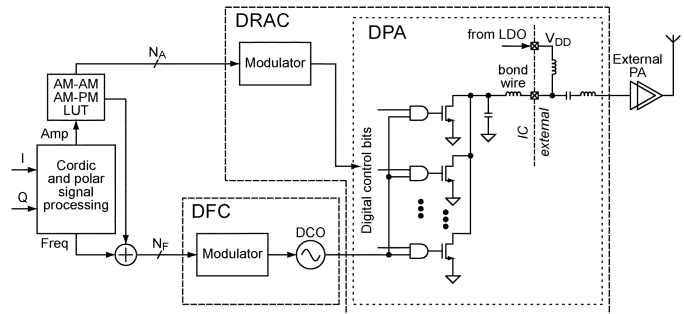


Fig. 1. A polar transmitter based on a DCO and DPA circuits from [2]. For simplicity, the all-digital PLL around the DCO is not shown.

a digital phase-modulated RF carrier by means of an all-digital PLL (ADPLL). The amplitude signal drives the N_A -bit *digital-to-RF-amplitude converter* (DRAC), which includes a digitally controlled power amplifier (DPA).

The DPA controls the envelope of the phase-modulated RF carrier; hence it is considered an RF-DAC. The DPA is different from the traditional RF-DACs in [3] or [4], because it does not use current sources. Therefore, the DPA is more compatible with low-voltage and low-cost digital CMOS processes than the traditional RF-DAC. Lack of current sources in the DPA results in a somewhat compressed transfer function, but the *look-up-table* (LUT) for AM-AM and AM-PM predistortion in the amplitude signal path shown in Fig. 1 linearizes the DPA transfer function.

References [1] and [2] have proved that the architecture in Fig. 1 is feasible for SoC meeting all GSM and EDGE specifications. However, the resolution of the amplitude modulation path is limited by lithography and RF mismatches (i.e., both amplitude and phase/delay) of the unit switching devices in the DPA, and, consequently, the polar transmitter has little margin in the far-out (i.e., the associated RX band) noise requirement of the SAW-less operation for EDGE.

The amplitude resolution could be improved by $\Sigma\Delta$ dithering of the unit transistor switches [1]–[3]. However, the quantization noise is pushed to higher frequencies where emission requirements might sometimes be difficult to satisfy, especially when considering radio coexistence in a wireless connectivity (e.g., Bluetooth, WLAN) or in a multicore RF-SoC environment.

In this paper, we address the far-out noise issue and make the digitally intensive polar architecture of Fig. 1 more attractive to multicore RF integration as well as to advanced modulation standards. We propose an RF-DAC structure in which significant resolution improvement is achieved by means of *incremental* pulswidth modulation (PWM). In other words, we will improve the resolution by adding a PWM-driven unit switching device on top of the existing RF-DAC structure instead of employing full-PWM. PWM is a *time-domain* operation since the signal is carried in width or duration of a pulse, which is time-

Manuscript received October 31, 2010; revised March 03, 2011 and March 11, 2011 accepted March 14, 2011. This paper was recommended by Associate Editor A. Neviani.

M. Park is with Maxim Integrated Products, Sunnyvale, CA 94086 USA (e-mail: Min.Park@maxim-ic.com).

M. H. Perrott is with the Masdar Institute of Science and Technology, Abu Dhabi, United Arab Emirates (e-mail: mhperrott@gmail.com).

R. B. Staszewski is with the Department of Microelectronics/Delft Institute of Microelectronics and Submicron Technology (DIMES), Faculty of Information Technology and Systems, Delft University of Technology, 2628 Delft, The Netherlands (e-mail: r.b.staszewski@tudelft.nl).

Digital Object Identifier 10.1109/TCSI.2011.2143030

domain information. In modern CMOS technology, time-domain processing can achieve higher resolution than voltage/current-domain processing because the switching time improves by device scaling [5]. Although the proposed amplitude resolution improvement is examined with a digital polar transmitter in this paper, it could be also applied to digitally intensive I/Q architecture.

In this paper, the PWM is categorized into *centered PWM* and *noncentered PWM* depending on the relative pulse position. The amplitude resolution improvement with centered PWM is presented in [6]. Noncentered PWM is easier to generate than centered PWM which requires pulse position adjustment to avoid phase distortion. In this work, we extend our discussion to the PWM scheme with noncentered PWM for its possible implementational advantage over centered PWM. Centered and noncentered PWM will be compared in the context of implementation and performance. It will be shown that the complexity and the resolution of a PWM generator, the complexity of amplitude/phase compensation, and the need for pulse positioning should be considered to choose between centered and noncentered PWM.

We first discuss the details of the PWM scheme to improve the amplitude resolution of the DPA. It will be shown that both centered and noncentered PWM create an incorrect RF signal when the pulsewidth is chosen in a straightforward way. Pre-distortion of PWM signals to generate a correct RF signal is then presented. LUT size reduction technique for the noncentered PWM scheme will then follow. Different polar transmitter architectures for centered PWM and noncentered PWM are proposed, and behavioral simulation results of each PWM scheme are presented to show the benefits of the proposed methods.

II. BACKGROUND: RESOLUTION ENHANCEMENT APPROACHES EMPLOYING PWM

We propose that the original DPA from Fig. 1 simply be augmented by one additional unit switching device that is driven by a PWM signal, as shown in Fig. 2. The amplitude resolution of the DPA improves by turning on the added switching device for only a short time interval within the positive half-cycle of the RF period. The RF output amplitude will be controlled by the time interval, and the resolution is determined by the time precision of the turning-on signal. The amplitude resolution of the DPA in Fig. 2 is now limited by the time resolution of the PWM. In modern CMOS processes, the switching time gets typically improved by $0.7\times$ per technology scaling node; hence, achieving higher resolution in the time-domain is easier than in the voltage/current-domain. In a 65 nm CMOS process, a minimum time resolution of 20 ps is easy to guarantee over process, temperature, and voltage variations. The switching devices in the original system are controlled by an integer word of the digital amplitude signal. In order to increase the amplitude resolution of the original DPA, a fractional word of the amplitude controls the extra switching device in Fig. 2.

The output amplitude of a PWM signal at the frequency of interest, however, is incorrect if the pulsewidth is chosen in a straightforward way such that the dc amplitude of the PWM signal is correct [7], [8]. This is in contrast with the normal up-conversion operation of the DPA, which acts as a mixer. As a result, this inaccurate RF output level at the carrier frequency turns out to limit the resolution improvement.

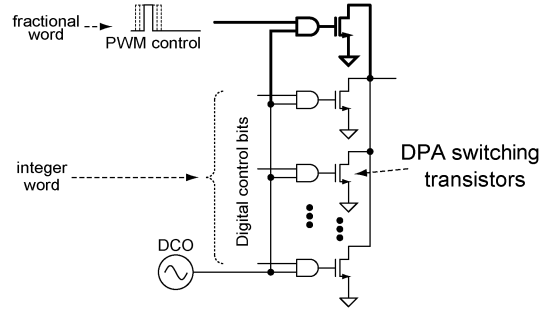


Fig. 2. Amplitude resolution improvement by adding a PWM driven transistor.

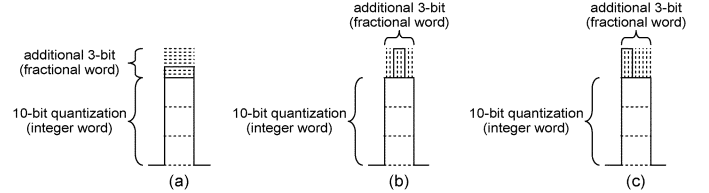


Fig. 3. Amplitude resolution improvement by (a) horizontal slicing; (b) vertical slicing with *centered PWM*; (c) vertical slicing with *noncentered PWM*.

To explain this issue, Fig. 3 illustrates three different quantization methods for adding an extra 3-bit resolution in either a voltage or current signal. For the DPA, the vertical axis in Fig. 3 represents the current or the conductance of the switching transistors, which is directly proportional to the output envelope, and its original resolution is 10 bits in the system of Fig. 1. Fig. 3(a) shows *horizontal slicing* of a signal, which is a conventional quantization method for a DAC. Both Fig. 3(b) and (c) show *vertical slicing* of a signal, where the output amplitude is controlled by the time interval of the vertically sliced signal. Note that the pulse in Fig. 3(b) is located at the center, whereas that in Fig. 3(c) is aligned at $t = 0$. This quantization method is PWM whose pulsewidth has a limited number of quantized pulsewidths. Depending on the position of the additive pulse, the vertical slicing method could be categorized as *centered PWM* and *noncentered PWM*, as shown in Fig. 3(b) and (c).

For the horizontal slicing scheme, the resolution is set by available area and power of switching transistors along with device mismatch, which limits the minimum device size. In contrast, the resolution of the vertical slicing scheme is set by the time-resolution. Accordingly, the vertical slicing can achieve higher resolution than the horizontal slicing with the same minimum device size if the time-resolution is finer. In modern nanometer-scale CMOS technology, the time-resolution is getting better, thus employing PWM seems a better choice to improve the amplitude resolution of a DPA.

Fig. 4 illustrates only the fractional portions from Fig. 3, where T is the time period of an RF carrier signal, v is a voltage/current level generated by the unit switching device using horizontal slicing, and τ is a pulsewidth for vertical slicing. Note that the amplitude switches are only turned on during half the DCO period, $T/2$. This is an important point as it realizes an implicit up-conversion mixer; hence no further mixing is required.

The full size of the voltage/current level generated by the unit switching device is normalized to 1. In this example, the fractional word adds an extra 3-bit resolution with 8 extra amplitude levels. The pulse position of the PWM signal could be at the center of the first half cycle of the carrier signal (centered

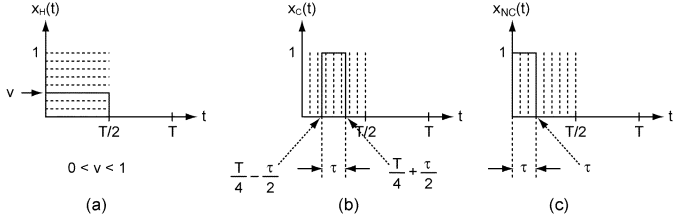


Fig. 4. The fractional portion of (a) horizontal slicing; (b) vertical slicing with centered PWM; (c) vertical slicing with noncentered PWM.

PWM), or it could be aligned with the main switching waveform (noncentered PWM).

Intuitively, all the signals from Fig. 4(a), (b), and (c) are the same in terms of power because the total area of the signals are the same. As pointed out in [7], however, they are equivalent only at dc. The Fourier transforms of the horizontal and vertical slicing signals are

$$X_H(j\omega) = \int_0^{T/2} v \cdot e^{-j\omega t} dt = v \cdot \frac{2e^{-j\frac{\omega T}{4}} \cdot \sin\left(\frac{\omega T}{4}\right)}{\omega}$$

$$X_C(j\omega) = \int_{\frac{T}{4}-\frac{\tau}{2}}^{\frac{T}{4}+\frac{\tau}{2}} 1 \cdot e^{-j\omega t} dt = \frac{2e^{-j\frac{\omega T}{4}} \cdot \sin\left(\frac{\omega \tau}{2}\right)}{\omega}$$

$$X_{NC}(j\omega) = \int_0^{\tau} 1 \cdot e^{-j\omega t} dt = \frac{2e^{-j\frac{\omega \tau}{2}} \sin\left(\frac{\omega \tau}{2}\right)}{\omega}$$

where $X_H(j\omega)$, $X_C(j\omega)$, and $X_{NC}(j\omega)$ are the Fourier transforms of the horizontal slicing, the centered PWM, and the noncentered PWM signals, respectively. In the examples of Fig. 4, v is $3/8$, and τ is $(3/8) \times (T/2)$. Fig. 5 illustrates the amplitudes and the phases of the Fourier transforms of each case, and it clearly shows that horizontal and vertical slicing *are not the same* at the carrier frequency, which is $1/T$ Hz. First of all, the amplitude of the Fourier transform of horizontal slicing is different from that of vertical slicing. Second of all, the phase of vertical slicing with noncentered PWM is different from that of horizontal slicing although the phase of centered PWM is the same as that of horizontal slicing. An RF-DAC is intended to generate a signal at a carrier frequency corresponding to an input digital code. Therefore, the vertical slicing signal in Fig. 5 creates an incorrect RF signal even though it creates an accurate dc signal.

An inaccurate RF signal from an RF-DAC leads to higher quantization noise. The Fourier transforms at a carrier frequency ($\omega = 2\pi/T$) should be examined for quantization noise analysis for an RF-DAC, and they are

$$X_H(j\omega)|_{\omega=2\pi/T} = \frac{-jvT}{\pi} \quad (1)$$

$$X_C(j\omega)|_{\omega=2\pi/T} = \frac{-j \sin\left(\frac{\pi\tau}{T}\right) T}{\pi} \quad (2)$$

$$X_{NC}(j\omega)|_{\omega=2\pi/T} = \frac{e^{-j\frac{\pi\tau}{T}} \sin\left(\frac{\pi\tau}{T}\right) T}{\pi}. \quad (3)$$

Fig. 6 depicts the amplitudes at the carrier frequency of the 8-level horizontal and vertical slicing signals, which are derived in (1) and (2) as a function of pulsewidth and horizontal slicing level. Note that the amplitudes of (2) and (3) are the same. The

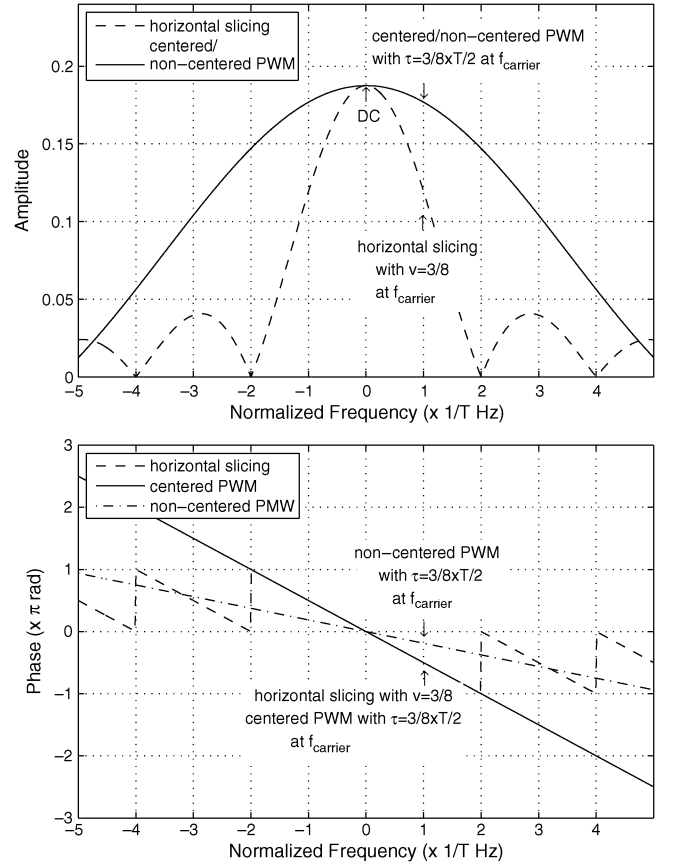


Fig. 5. Frequency domain amplitude/phase comparison of horizontal slicing and vertical slicing (centered PWM and noncentered PWM) when $v = 3/8$ and $\tau = (3/8) \times (T/2)$.

maximum amplitude at the carrier frequency is normalized to 1 for simple quantization error estimation, and Table I shows the RF amplitude for each input code and the corresponding quantization error. In Table I, the range of the quantization error of vertical slicing is much larger than $1/16 = 0.0625$. It means that the quantization noise by vertical slicing is increased relative to that of horizontal slicing. The quantization noise power can be estimated by a probability density function assuming the probability of error is uniformly distributed. The calculated noise power shows that horizontal slicing achieves 3-bit resolution while vertical slicing achieves only 0.9-bit resolution. Therefore, resolution improvement by PWM is severely impaired if the pulsewidth is chosen in such a way that the dc amplitude of a vertical slicing signal is matched with that of a horizontal slicing signal, as shown in Fig. 6 and Table I.

Vertical slicing with noncentered PWM distorts the signal even further due to its phase discrepancy. The phase of the Fourier transforms in (1) and (2) is $-(\pi/2)$ regardless of v or τ , but the phase in (3) is dependent on τ . Accordingly, the phase of the Fourier transforms of the horizontal slicing signals and the noncentered PWM signals are different unless τ is $T/2$. Thus, the phase of noncentered PWM should be compensated for the correct symbol representation.

The importance of phase compensation is presented in Fig. 7. It illustrates the trajectories of an example of EDGE samples in an I/Q plane with and without phase error. Fig. 7(a) shows the ideal symbol trajectory without any amplitude/phase quantization. Fig. 7(b) shows the symbol trajectory with amplitude

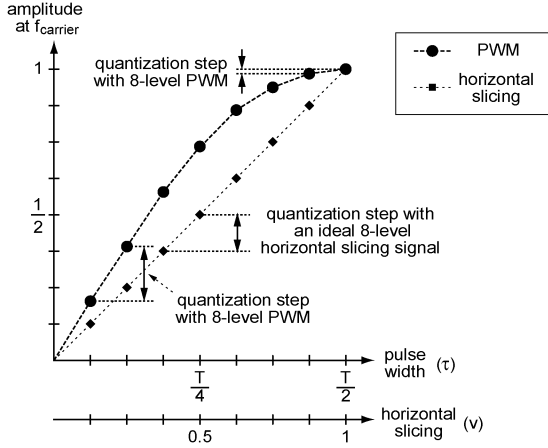


Fig. 6. Amplitude quantization step at the carrier frequency of horizontal slicing signals and 8-level PWM signals [8].

TABLE I
QUANTIZATION ERROR BY HORIZONTAL SLICING AND VERTICAL SLICING
(CENTERED/NONCENTERED PWM)

original input	pulse width	horizontal slicing		vertical slicing	
		amp at f_c	quantization error	amp at f_c	quantization error
$0 \sim \frac{1}{16}$	0	0	$-\frac{1}{16} \sim 0$	0	$-\frac{1}{16} \sim 0$
$\frac{1}{16} \sim \frac{3}{16}$	$\frac{1}{8} \cdot \frac{T}{2}$	$\frac{1}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.194	$0.132 \sim 0.007$
$\frac{3}{16} \sim \frac{5}{16}$	$\frac{2}{8} \cdot \frac{T}{2}$	$\frac{2}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.383	$0.195 \sim 0.070$
$\frac{5}{16} \sim \frac{7}{16}$	$\frac{3}{8} \cdot \frac{T}{2}$	$\frac{3}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.556	$0.243 \sim 0.118$
$\frac{7}{16} \sim \frac{9}{16}$	$\frac{4}{8} \cdot \frac{T}{2}$	$\frac{4}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.707	$0.270 \sim 0.145$
$\frac{9}{16} \sim \frac{11}{16}$	$\frac{5}{8} \cdot \frac{T}{2}$	$\frac{5}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.832	$0.269 \sim 0.144$
$\frac{11}{16} \sim \frac{13}{16}$	$\frac{6}{8} \cdot \frac{T}{2}$	$\frac{6}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.924	$0.236 \sim 0.111$
$\frac{13}{16} \sim \frac{15}{16}$	$\frac{7}{8} \cdot \frac{T}{2}$	$\frac{7}{8}$	$-\frac{1}{16} \sim \frac{1}{16}$	0.981	$0.168 \sim 0.043$
$\frac{15}{16} \sim \frac{16}{16}$	$\frac{8}{8} \cdot \frac{T}{2}$	$\frac{8}{8}$	$-\frac{1}{16} \sim 0$	1	$-\frac{1}{16} \sim 0$

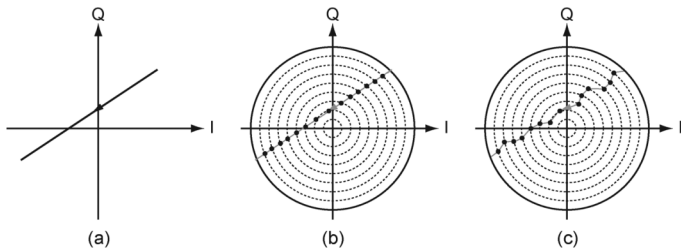


Fig. 7. Symbol trajectories for an EDGE signal in I/Q plane (a) ideal trajectory; (b) quantized trajectory without amplitude or phase error; (c) quantized trajectory with phase error.

quantization. There is no phase error in this case so that its spectrum will show only the noise due to the amplitude quantization error. Horizontal slicing and centered PWM signals will show this symbol trajectory. Fig. 7(c) shows the trajectory with amplitude quantization and phase error. Even though it has the same amplitude quantization as Fig. 7(b), the phase error results in uneven symbol trajectory, which will cause more noise in the spectrum.

It is worth mentioning that the far-out noise of the system in Fig. 1 is dominated by the quantization noise of the amplitude signal of the DPA. The high frequency component due to

switching action is filtered by the simple off-chip LC filter used for *near-class-E* operation of the DPA [1]. The ripple frequency from the PWM signal is much higher than the switching frequency of the DPA; thus, the ripple signal does not contribute to the far-out noise of interest (i.e., the noise at the associated RX band). We will focus on lowering the amplitude quantization noise to decrease the far-out noise of the polar transmitter.

III. PREDISTORTION OF PWM SIGNALS

Amplitude distortion of PWM exists for both centered and noncentered PWM. In contrast, phase distortion is caused by only noncentered PWM. We will first discuss how to avoid or reduce amplitude distortion of centered PWM, which was briefly introduced in [6], and then discuss phase and amplitude distortion of noncentered PWM.

A. Predistortion of PWM Signal for Centered PWM

One solution to the amplitude discrepancy between horizontal slicing and centered PWM is choosing the pulsewidth for the vertical slicing such that its amplitude at the carrier frequency is the same as that of corresponding horizontal slicing

$$X_H(j\omega)|_{\omega=\frac{2\pi}{T}} = X_C(j\omega)|_{\omega=\frac{2\pi}{T}}$$

$$\frac{-jvT}{\pi} = \frac{-j \sin\left(\frac{\pi\tau}{T}\right)T}{\pi}$$

$$v = \sin\left(\frac{\pi\tau}{T}\right). \quad (4)$$

Equation (4) shows how to choose the pulsewidth for the centered PWM signal. A centered PWM signal whose pulsewidth satisfies (4) has the same amplitude as the corresponding horizontal slicing signal at the carrier frequency. Note that the centered PWM signal has the same phase as that of the horizontal slicing signal. Thus, phase distortion is not a concern for a centered PWM signal.

Fig. 8 depicts an example of the amplitude of the Fourier transform of the centered PWM signal satisfying (4). In Fig. 8, it is clear that the amplitude of the centered PWM signal at the carrier frequency is the same as that of the horizontal slicing signal. Table II shows an example of the pulsewidths, for which the amplitude and the phase at the carrier frequency is the same as those of the corresponding horizontal slicing signals. The amplitude resolution of the centered PWM signals employing the pulsewidths shown in Table II is also the same as that of horizontal slicing signals. Unfortunately, the pulsewidths shown in Table II are very challenging to generate without an accurate delay controller, such as a high precision DLL or a fine-resolution delay line. Therefore, such PWM generation seems impractical.

A PWM signal is easier to generate when its pulsewidth is integer multiples of a certain stable delay. Table III shows an example in which the pulsewidths are integer multiples of $T/16$. It also presents the corresponding horizontal slicing signals that satisfy (4). A simple delay chain can generate those pulsewidths shown in Table III; hence, it is a better implementation choice. However, the relationship between the desired horizontal slicing signals and the pulsewidths shown in Table III is nonlinear. A predistortion digital LUT can be utilized to implement this mapping, where the contents of the predistortion LUT should be

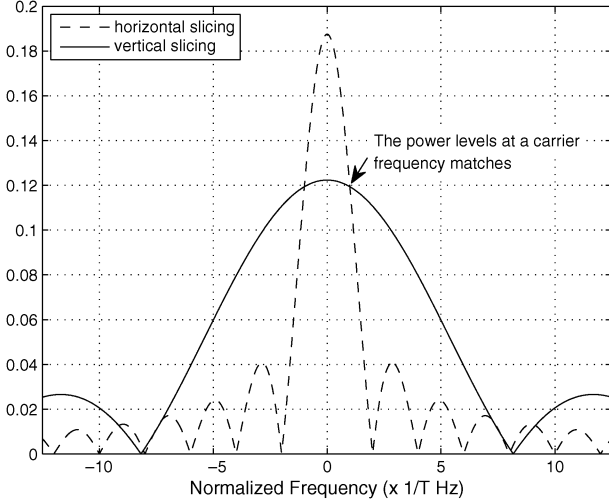


Fig. 8. Frequency domain amplitude comparison of horizontal slicing and centered PWM when $v = 3/8$ and $\tau = 0.122 \times T$.

TABLE II
PULSEWIDTHS THAT CREATE THE SAME AMPLITUDE AT A CARRIER FREQUENCY AS THE CORRESPONDING HORIZONTAL SLICING SIGNALS

horizontal slicing	pulsewidth	horizontal slicing	pulsewidth
0	0	$\frac{4}{8}$	$0.167 \times T$
$\frac{1}{16}$	$0.0399 \times T$	$\frac{5}{8}$	$0.215 \times T$
$\frac{2}{16}$	$0.0804 \times T$	$\frac{6}{8}$	$0.270 \times T$
$\frac{3}{16}$	$0.122 \times T$	$\frac{7}{8}$	$0.339 \times T$

TABLE III
PULSEWIDTHS, WHICH ARE THE MULTIPLES OF $T/16$, AND THE EQUIVALENT HORIZONTAL SLICING SIGNALS

horizontal slicing	pulsewidth	horizontal slicing	pulsewidth
0	0	0.707	$\frac{4}{8} \times \frac{T}{2}$
0.194	$\frac{1}{8} \times \frac{T}{2}$	0.831	$\frac{5}{8} \times \frac{T}{2}$
0.383	$\frac{2}{8} \times \frac{T}{2}$	0.924	$\frac{6}{8} \times \frac{T}{2}$
0.556	$\frac{3}{8} \times \frac{T}{2}$	0.981	$\frac{7}{8} \times \frac{T}{2}$

chosen such that the amplitude quantization error is minimized.

Table IV presents an example of a predistortion digital LUT for mapping the input code to the appropriate pulsewidth of centered PWM. In the example of Table IV, the LUT maps the 5-bit input to 8-level pulsewidths. The LUT mapping linearizes the relationship between the input code and the output amplitude at the carrier frequency such that the amplitude quantization error is minimized. Note that the range of the quantization errors in Table IV are sometimes larger than the maximum quantization error of the ideal 8-level signal, which is $\pm 1/16 = \pm 0.0625$. Fig. 6 explains the reason of the larger quantization noise of the LUT in Table IV more clearly. The quantization step is larger than that of ideal 8-level horizontal slicing when the pulsewidths are relatively short, as shown in Fig. 6. Therefore, the quantization error using the predistortion LUT and 8-level centered PWM will be larger than that of the ideal 3-bit horizontal slicing. The quantization noise power of the centered PWM employing

TABLE IV
LOOK-UP-TABLE EXAMPLE WITH 5-BIT INPUT AND 8-LEVEL CENTERED PWM OUTPUT

original input	LUT input	pulsewidth selected	amp at f_c	quantization error
$0 \sim \frac{1}{64}$	0	0	0	$0 \sim 0.0156$
$\frac{1}{64} \sim \frac{3}{64}$	$\frac{1}{32}$	0	0	$0.0156 \sim 0.0469$
$\frac{3}{64} \sim \frac{5}{64}$	$\frac{2}{32}$	0	0	$0.0469 \sim 0.0781$
$\frac{5}{64} \sim \frac{7}{64}$	$\frac{3}{32}$	0	0	$0.0781 \sim 0.109$
$\frac{7}{64} \sim \frac{9}{64}$	$\frac{4}{32}$	$\frac{1}{8} \cdot \frac{T}{2}$	0.194	$-0.0857 \sim -0.0545$
$\frac{9}{64} \sim \frac{11}{64}$	$\frac{5}{32}$	$\frac{1}{8} \cdot \frac{T}{2}$	0.194	$-0.0545 \sim -0.0232$
$\frac{11}{64} \sim \frac{13}{64}$	$\frac{6}{32}$	$\frac{1}{8} \cdot \frac{T}{2}$	0.194	$-0.0232 \sim 0.00804$
$\frac{13}{64} \sim \frac{15}{64}$	$\frac{7}{32}$	$\frac{1}{8} \cdot \frac{T}{2}$	0.194	$0.00804 \sim 0.0393$
$\frac{15}{64} \sim \frac{17}{64}$	$\frac{8}{32}$	$\frac{1}{8} \cdot \frac{T}{2}$	0.194	$0.0393 \sim 0.0705$
$\frac{17}{64} \sim \frac{19}{64}$	$\frac{9}{32}$	$\frac{1}{8} \cdot \frac{T}{2}$	0.194	$0.0705 \sim 0.102$
$\frac{19}{64} \sim \frac{21}{64}$	$\frac{10}{32}$	$\frac{2}{8} \cdot \frac{T}{2}$	0.383	$-0.0858 \sim -0.0546$
$\frac{21}{64} \sim \frac{23}{64}$	$\frac{11}{32}$	$\frac{2}{8} \cdot \frac{T}{2}$	0.383	$-0.0546 \sim -0.0233$
	\vdots	\vdots		
$\frac{53}{64} \sim \frac{55}{64}$	$\frac{27}{32}$	$\frac{5}{8} \cdot \frac{T}{2}$	0.831	$-0.00335 \sim 0.0279$
$\frac{55}{64} \sim \frac{57}{64}$	$\frac{28}{32}$	$\frac{5}{8} \cdot \frac{T}{2}$	0.831	$0.0279 \sim 0.0592$
$\frac{57}{64} \sim \frac{59}{64}$	$\frac{29}{32}$	$\frac{6}{8} \cdot \frac{T}{2}$	0.924	$-0.0333 \sim -0.00201$
$\frac{59}{64} \sim \frac{61}{64}$	$\frac{30}{32}$	$\frac{6}{8} \cdot \frac{T}{2}$	0.924	$-0.00201 \sim 0.0292$
$\frac{61}{64} \sim \frac{63}{64}$	$\frac{31}{32}$	$\frac{7}{8} \cdot \frac{T}{2}$	0.981	$-0.0277 \sim 0.00359$
$\frac{63}{64} \sim \frac{64}{64}$	$\frac{31}{32}$	$\frac{7}{8} \cdot \frac{T}{2}$	0.981	$-0.0156 \sim 0$

the LUT in Table IV can be also calculated by probability density functions, and the calculated amplitude resolution at the carrier frequency is 2.6 bits.

As we have seen in Tables II and III, there are two options to linearize the RF power of a PWM signal: mapping uniform data input to nonuniform pulsewidths (Table II), and mapping nonuniform data input to uniform pulsewidths (Table III). Apparently, uniform-to-nonuniform mapping needs no sacrifice for achieving 3-bit resolution, while nonuniform-to-uniform mapping shows the resolution degradation by 0.4 bits. However, the pulsewidths in Table III are easy to generate using a simple delay chain. The predistortion LUT is also easy to implement in digital CMOS processes. Therefore, we propose to employ a predistortion LUT for centered PWM, which helps to minimize the overall system complexity with slight degradation of amplitude resolution.

B. Predistortion of PWM Signals for Noncentered PWM

In contrast to the centered PWM, phase compensation is required for noncentered PWM as the phase of the Fourier transforms of the horizontal slicing signals and the noncentered PWM signals are different. Therefore, the phase error created by a noncentered PWM signal should be compensated. The proposed PWM scheme is applied to a digital polar transmitter including a PLL, which can manipulate the phase of an RF signal. The PLL can compensate the phase error, but it requires an additional LUT for the phase signal path.

When considering the full RF-DAC output, not only the phase but also the amplitude of the Fourier transform of noncentered PWM is different from that of centered PWM. Equations (1),

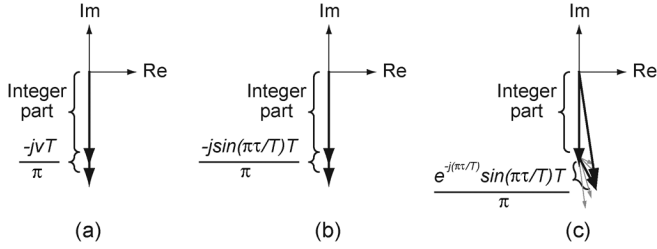


Fig. 9. Vectors of (a) an integer part of the amplitude signal and a horizontal slicing's fractional part; (b) an integer part and a centered PWM's fractional part; (c) an integer part and a noncentered PWM's fractional part.

(2), and (3) are the Fourier transforms of the fractional part of the amplitude signal only. The Fourier transforms of the complete signals including both the integer and the fractional portions are

$$X_{H,total}(j\omega)|_{\omega=2\pi} = \frac{-jVT}{\pi} + \frac{-jvT}{\pi} \quad (5)$$

$$X_{C,total}(j\omega)|_{\omega=2\pi} = \frac{-jVT}{\pi} + \frac{-j \sin(\frac{\pi\tau}{T}) T}{\pi} \quad (6)$$

$$X_{NC,total}(j\omega)|_{\omega=2\pi} = \underbrace{\frac{-jVT}{\pi}}_{\text{integer part}} + \underbrace{\frac{e^{-j\frac{\pi\tau}{T}} \sin(\frac{\pi\tau}{T}) T}{\pi}}_{\text{fractional part}} \quad (7)$$

where $X_{H,total}(j\omega)$, $X_{C,total}(j\omega)$, and $X_{NC,total}(j\omega)$ are the Fourier transforms of the horizontal slicing, the centered PWM, and the noncentered PWM signals, respectively, including both the integer and fractional part. V is the integer portion, which is generated by a DPA. The phases of the integer part and the fractional part in (5) and (6) are the same. Therefore, if the fractional portion of a centered PWM signal in (6) is the same as that of the targeted horizontal slicing signal in (5), then the amplitudes of the complete signals are the same. In other words, τ for centered PWM is only dependent on v not V . However, as shown in (7), the phase of the fractional part of a noncentered PWM signal is different from that of the integer part, and the amplitude of the final signal is also different due to the vector sum of the integer and fractional parts in complex plane. Fig. 9 illustrates the situation more clearly. The goal of choosing the appropriate pulsewidth τ for centered and noncentered PWM is to make the total signal including the integer portion the same as the ideal horizontal slicing signal, which is depicted in Fig. 9(a). For centered PWM, it is a simple task because the pulsewidth τ is determined by only v , and it is independent of the integer portion V . For noncentered PWM, however, the phase of the fractional portion is not the same as that of the integer portion; as a result, the amplitude of the total signal is a vector sum of the integer and the fractional parts, as shown in Fig. 9(c). The vector sum is a function of the integer part as well as the pulsewidth τ . Therefore, the amplitude predistortion LUT for noncentered PWM should take the integer part into consideration while the LUT for centered PWM only takes the fractional part as its input. In other words, the size of the amplitude predistortion LUT for noncentered PWM should be bigger than the LUT for centered PWM.

As an example, the full-length size of the amplitude predistortion LUT for the noncentered PWM should be 32768 words (15-bit = 10-bit + 5-bit) if the integer word of the amplitude is 10 bits and the fractional word is 5 bits. However, the size of the amplitude predistortion LUT for centered PWM will be only 32

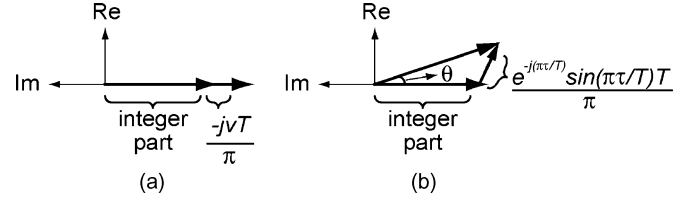


Fig. 10. Vectors of (a) horizontal slicing; (b) noncentered PWM in complex plane. The entire complex plane is rotated 90° counterclockwise.

words (5-bit) because the LUT takes only the fractional part as an input. Accordingly, noncentered PWM requires much bigger amplitude predistortion LUT than centered PWM.

Nevertheless, under certain conditions, the size of the amplitude predistortion LUT for noncentered PWM can be significantly reduced by approximation. According to (7), the integer part is always imaginary, and the phase of the fractional part has both real and imaginary parts that depend on τ . Note that the real part comes only from the fractional word. Equation (7) can be separated into the real and imaginary part, and the real part can be neglected if the integer portion, V , is much larger than the fractional portion, as follows:

$$X_{NC,total}(j\omega)|_{\omega=2\pi} = \frac{-jVT}{\pi} - \frac{j \sin(\frac{\pi\tau}{T}) \sin(\frac{\pi\tau}{T}) T}{\pi} + \frac{\cos(\frac{\pi\tau}{T}) \sin(\frac{\pi\tau}{T}) T}{\pi}$$

$$X_{NC,total}(j\omega)|_{\omega=2\pi} \approx \frac{-jVT}{\pi} - \frac{j \sin(\frac{\pi\tau}{T}) \sin(\frac{\pi\tau}{T}) T}{\pi} \quad \text{if } V \gg \cos\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right). \quad (8)$$

Equation (8) is valid if V is large enough. In Fig. 10, where θ is the phase of the vector of noncentered PWM signal, we can easily see that θ will become very small as V , which represents the integer part of the baseband signal, increases such that the real part becomes negligible compared to the imaginary part.

Therefore, (8) is a very good approximation for the large amplitude signals. For an EDGE signal, the amplitude is always larger than certain amplitude in order to keep its peak-to-minimum ratio of signal around 16.4 dB. As such, V might be large enough to satisfy the condition for (8) if the baseband input signal is an EDGE signal. Validity of (8) for an EDGE signal is verified by simulations and the results will be presented in Section V-B. If (8) is valid, the same size of the amplitude predistortion LUT as centered PWM can be employed for amplitude predistortion of noncentered PWM. The LUT should be realized based on the following equation:

$$X_H(j\omega)|_{\omega=2\pi} = X_{NC}(j\omega)|_{\omega=2\pi}$$

$$\therefore v = \sin\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right). \quad (9)$$

Furthermore, Fig. 9(c) indicates that not only an amplitude predistortion LUT but also a phase compensation LUT should take the integer part of the amplitude into account because the phase of the noncentered PWM signal depends on the integer word as well as τ . As a result, the size of the phase predistortion LUT will be huge. However, some approximation technique can help to reduce its size, too.

Fig. 10 compares an ideal horizontal slicing signal and a noncentered PWM counterpart. Note that the entire complex plane

is rotated by 90° counterclockwise for easy \tan^{-1} approximation. The amount of phase distortion by noncentered PWM is θ , which is

$$\theta = \tan^{-1} \left(\frac{\cos\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)}{V + \sin\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)} \right). \quad (10)$$

Thus, the amount of phase predistortion should be

$$\phi = -\tan^{-1} \left(\frac{\cos\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)}{V + \sin\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)} \right) \quad (11)$$

where ϕ is the required phase predistortion for a noncentered PWM signal. Approximation of \tan^{-1} by Taylor series is

$$\tan^{-1}(x) \approx x \quad \text{if } x \text{ is close to } 0. \quad (12)$$

Since θ converges to 0 if the integer part increases, as shown in Fig. 10(b), (12) is valid when the integer part is large enough. With this approximation, (11) becomes a simple closed-form expression as follows:

$$\begin{aligned} \phi &\approx -\frac{\cos\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)}{V + \sin\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)} \\ &\approx -\frac{\cos\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right)}{V} = -\frac{1}{V} \cdot \cos\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right) \\ &\quad \text{if } V \gg \sin\left(\frac{\pi\tau}{T}\right) \sin\left(\frac{\pi\tau}{T}\right). \end{aligned} \quad (13)$$

Equation (13) implies that the phase predistortion LUT can be broken into two smaller LUTs and a multiplier. One LUT takes the integer word of the amplitude signal and calculates $1/V$ while the other LUT takes the fractional word and calculates $\cos(\pi\tau/T) \sin(\pi\tau/T)$. The total amount of the phase compensation is the multiplication of those two outputs. In this way, the phase predistortion LUT can be implemented with less complexity. In a polar transmitter, the LUT to compensate AM-PM distortion of RF PA is likely to already exist in the system. In this case, the LUT for $1/V$ can be combined with the existing AM-PM predistortion LUT, and only a small LUT with the fractional word inputs calculating $\cos(\pi\tau/T) \sin(\pi\tau/T)$ and a multiplier are required additionally.

Validity of (13) is confirmed, as shown in Fig. 11. The integer word is 10 bits, which ranges from 0 to 1023, and the fractional word is 5 bits (32 levels). The approximation in (13) agrees well with (11) when the input code is over 100 although they differ when the input code is less than 6. Note that the input code represents the amplitude of an input baseband signal in a polar transmitter although we are currently dealing with phase compensation for noncentered PWM. For an EDGE signal, the input code is always larger than certain amplitude in order to keep its peak-to-minimum ratio, as explained earlier. As a result, (13) is a very good approximation for EDGE, and it will be verified by simulations in Section V-B.

The resolution of the phase path of a polar transmitter is another restriction to affect complexity of the entire system. The size of a phase predistortion LUT depends on not only the input resolution but also the output resolution, which goes into the phase input of a PLL. If an ADPLL is employed, the resolution of the phase input of an ADPLL also affects complexity of its building blocks, such as a DCO, a digital loop filter, and a TDC.

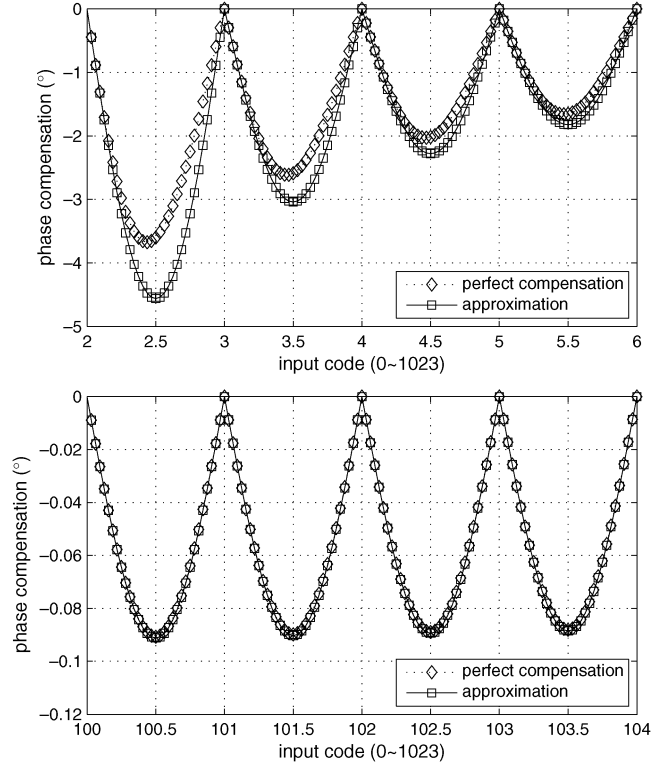


Fig. 11. Phase compensation comparison between (11) and (13) for small (top) and large (bottom) values of the input code.

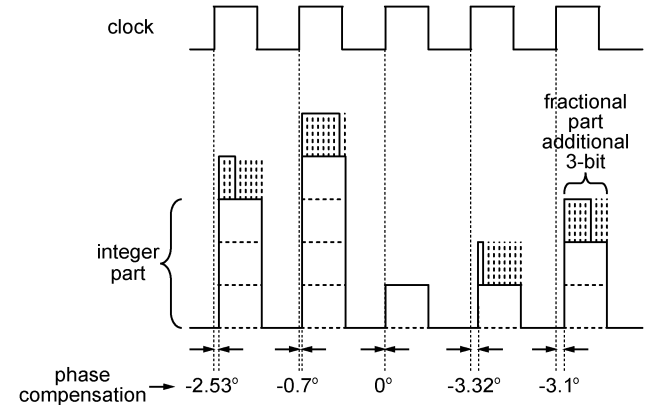


Fig. 12. An example of timing diagrams of phase compensation for noncentered PWM.

The required phase resolution of the noncentered PWM method can be determined by simulations, and the results with different phase resolutions are presented in Section V-B.

Note that the phase compensation for noncentered PWM should occur instantly (i.e., having wide enough bandwidth) as illustrated in Fig. 12. It means that an ADPLL should employ a special method such as a two-point modulation. [9] employs a feed-forward path for an instant phase/frequency shift, and also uses a predistortion path to compensate closed-loop counteract; hence, the instant phase shift shown in Fig. 12 is possible.

IV. AMPLITUDE RESOLUTION IMPROVEMENT EMPLOYING PWM

Given the previous analysis, the architecture for improving the amplitude resolution of a polar transmitter employing a

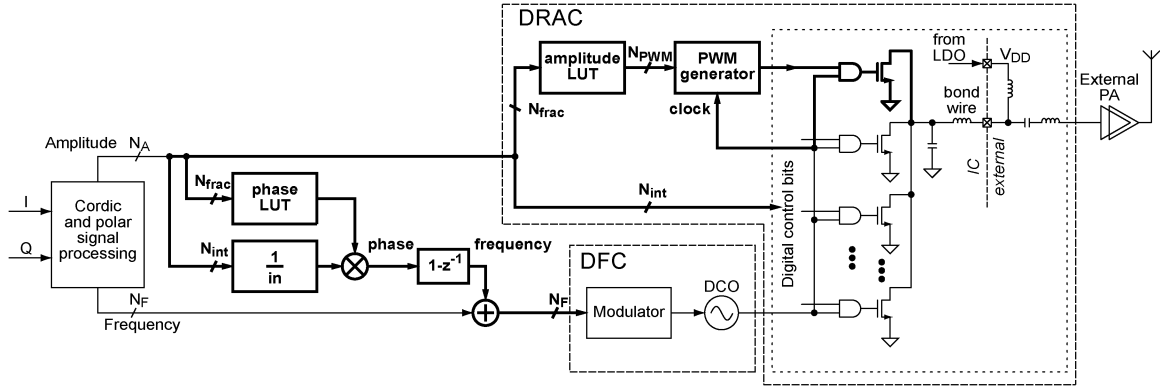


Fig. 13. Proposed architecture with noncentered PWM.

PWM generator will be proposed. The architecture for centered and noncentered PWM should differ because of their different characteristics. The polar transmitter architecture employing centered PWM and its detailed explanation can be found in [6]. In this paper, the architecture based on centered PWM will be omitted. We will start by introducing the polar transmitter employing noncentered PWM. Then, a simple delay chain circuit, which can be used in a PWM generator, will be also proposed and its circuit-level simulation results will be shown. Finally, centered and noncentered PWM methods will be compared.

A. Proposed Architecture Employing Noncentered PWM

Fig. 13 depicts the proposed architecture with a noncentered PWM generator. The proposed architecture is augmented from the original polar transmitter architecture in Fig. 1. In Fig. 13, the PWM generator makes $2^{N_{\text{PWM}}}$ -level noncentered PWM signals. The phase LUT takes the N_{frac} -bit fractional word of the amplitude signal as an input, and it calculates $\cos(\pi\tau/T) \sin(\pi\tau/T)$, as per (13). The N_{int} -bit integer word of the amplitude signal goes to the functional block, which calculates the inverse. It could be an arithmetic logic or an LUT. In most commercial polar transmitter systems, there is an AM-PM LUT. Hence, the $1/in$ block can be simply combined with the existing LUT. The combination of the phase LUT and the $1/in$ block calculates the approximated phase predistortion value, as explained earlier. The additional LUT for amplitude path is also used, and it takes the fractional word of the amplitude signal. The content of the LUT is derived according to (9), which is an approximation. Note that the *Pulse Center* block in [6] is unnecessary in this case because noncentered PWM is always aligned with the main clock as far as the PWM generator is referenced to the main clock.

Fig. 14 shows the conceptual block diagram of a noncentered PWM generator, and an example timing diagram. In this case, the generated PWM is aligned with the rising edge of the main clock. If necessary, the functional block in Fig. 14 can be easily modified such that the PWM is aligned with the falling edge of the clock instead of the rising edge. Then, the content of the amplitude and phase predistortion LUTs should be changed, but the overall architecture will remain the same. Note that Fig. 14 requires only one delay chain. Compared to the centered PWM generator in [6], therefore, it is much simpler overall to implement. In terms of the mismatch effect of the delay chain, the noncentered PWM generator proposed is affected by only one

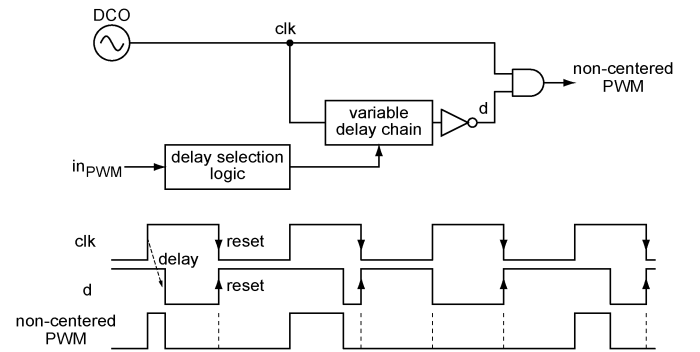


Fig. 14. The conceptual block diagrams of a noncentered PWM generator and the timing diagrams.

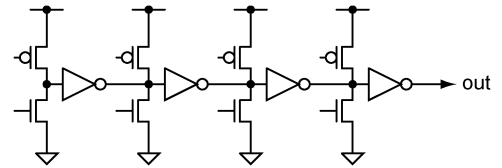


Fig. 15. The proposed delay chains for a PWM generator.

delay chain, whereas the centered PWM one is affected by two independent delay chains [6].

B. Proposed Delay Chain Circuit

The PWM generator in Fig. 14 requires delay chains, in which the delay is controllable digitally. A simple inverter-based delay chain for the PWM generator is proposed in this section. Fig. 15 illustrates the proposed circuit. The inverters are cascaded, and their input nodes can be set to the desired logic levels by force in such a way that some inverters are held to a certain voltage while the others propagate signals from the previous stages. There are 3 possible states for each input node of the inverters: high, low, and local floating. When the input nodes of the inverters are set to floating, the voltage of that node is determined by the previous stage; thus, that inverter propagates a digital signal. By controlling the number of the inverters propagating a signal, one can change the total propagation delay of the circuit.

Fig. 16 shows how the proposed circuit creates 4 inverter delays. In the reset state, the inputs of all 4 inverters are held at a certain voltage by *preset devices* at the input nodes of the inverters such that the delay chain's output is low. In order to generate 4 inverter delays, the inputs of the last 3 inverters are set to floating. Now, the first stage determines the output. Therefore,

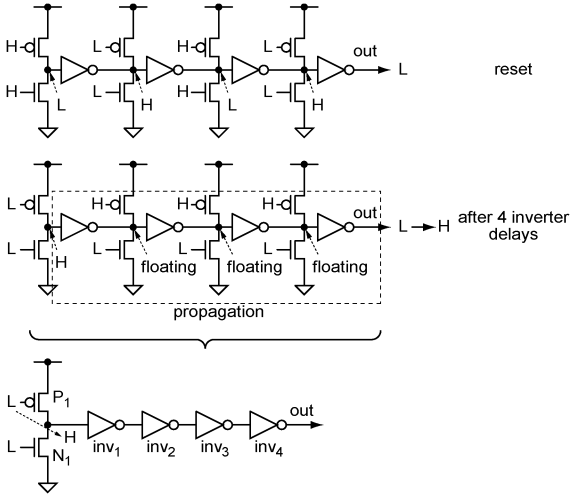


Fig. 16. Generation of 4 inverter delays.

after the input of the first inverter, inv_1 , goes to high, the signal propagates through the 4 inverters. The total delay is

$$delay = t_{clk-to-start1} + 4 \cdot delay_{inv}$$

where $delay_{inv}$ is an inverter delay, and $t_{clk-to-start1}$ is the time to take for the input of inv_1 to become high, which is the delay from the clock to the start of the signal. $t_{clk-to-start1}$ is determined by the size of one of the preset devices, P_1 , and the total capacitance of the input node of inv_1 .

Fig. 17 shows the case of 3 inverter delays. In order to create 3 inverter delays, the inputs of the last 2 inverters, inv_3 and inv_4 , become floating. When the input of the first 2 inverters, inv_1 and inv_2 , are set to high and low, respectively, the signal propagates through the last 3 inverters creating 3 inverter delays. Note that the input of inv_1 is still low before $t_{clk-to-start1}$. Thus, inv_1 attempts to make the output, which is the input of inv_2 , high before $t_{clk-to-start1}$ while N_2 tries to make the input of inv_2 low. If N_2 is much larger than the size of inv_1 , then the input of inv_2 will become low by force even before $t_{clk-to-start1}$. In this case, $t_{clk-to-start2}$, the time to take for the input of inv_2 to become low, will be determined by the size of N_2 and the input node capacitance of inv_2 . If the size of N_2 is small, then $t_{clk-to-start2}$ will be longer than $t_{clk-to-start1}$ because the input of inv_2 can become low only after the input of inv_1 becomes high. In any case, the total delay of Fig. 17 is

$$delay = t_{clk-to-start2} + 3 \cdot delay_{inv}$$

Employing smaller inverters relative to the preset devices decreases the clock-to-start time, but increases the inverter delay. This trade-off should be taken into account for transistor sizing of the proposed delay chain circuit.

The final propagation delay value of the proposed delay chain circuit is controlled by digital signals. Hence, the digital circuits that generate the proper control signals for each stage are required. Moreover, those signals should be referenced to the clock in that the output of the delay chain should be a *delayed* clock. Fig. 18 depicts the details of the delay generator including the digital circuits to create the control signals for each stage. Note that AND and OR gates with one input fed by the clock are required so that all the control signals are aligned with the

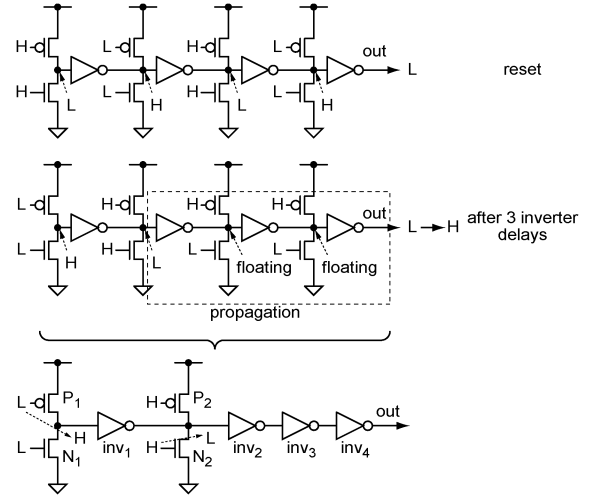


Fig. 17. Generation of 3 inverter delays.

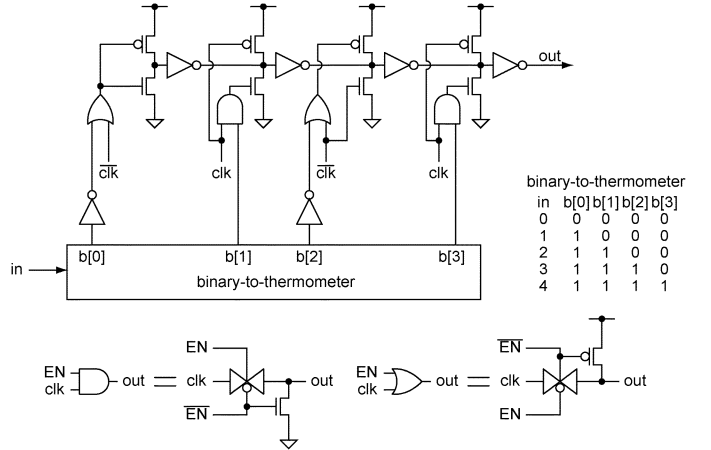


Fig. 18. The proposed delay chains circuits including control logic.

clock. In addition, both clk and \overline{clk} are required, as shown in Fig. 18. \overline{clk} is usually available in PLLs using a differential LC oscillator.

One drawback of the proposed circuit is that the loading for the clock signal gets bigger and bigger as the number of inverter stages increases. However, the main clock is also supposed to drive the DPA, which is composed of 1024 equivalent unit switching transistors [1], [2]. Compared to the clock loading by the switching transistors, the additional loading caused by the delay stage is relatively small. Therefore, the additional clock loading will not require major design modification of the conventional DPA circuit.

The post layout simulations of the proposed delay chain circuit are done in a 65 nm CMOS process. The total number of chain inverters designed is 25, and the circuit simulation results will be applied to the behavioral-level simulations of the proposed system in the next section. Ideally, the delay created by the proposed circuit is an integer multiple of one inverter delay plus a clock-to-start time, as explained earlier. However, the delay of each inverter will be different because of device mismatch. The delay variation of the inverters is estimated by 500 runs of Spectre Monte Carlo simulations. The estimated delay variation will also be taken into account for the behavioral simulations of the proposed architecture in the next section.

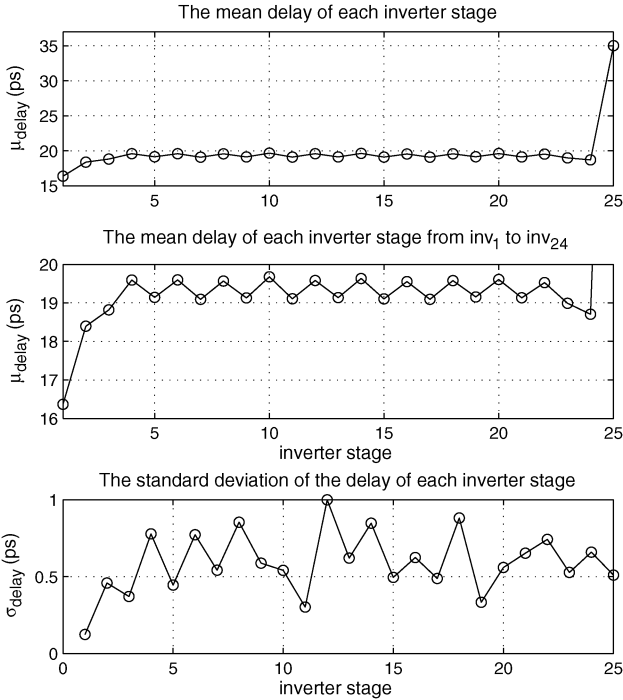


Fig. 19. The mean and standard deviation of the inverter stage delay ($inv_1 \sim inv_{24}$).

Fig. 19 depicts the mean delays and the standard deviations of the inverter stage. It shows the incremental delay by each inverter stage, and the delay of inv_{25} includes the contribution of a clock-to-start time. By enabling more inverters, the total propagation delay will increase in steps of around 19.5 ps, which is one inverter delay. However, the minimum propagation delay of the circuit is 35 ps because the minimum delay is one inverter delay plus a clock-to-start delay. Fig. 19 also zooms in the mean delay of inv_1 to inv_{24} . It shows that the delay goes slightly up and down alternatively as the input code increases. This is because of the delay mismatch of rise and fall times of the inverters. The standard deviation of each delay is less than 1 ps, which is about 5% of the nominal delay, and this result will be used for the behavioral simulations in the next section.

The jitter from the inverter chain will add the noise to both the pulse position and the pulsewidth, and it is an additional noise source for PWM. However, the PWM signal drives only one unit switching device. In a 10-bit RF-DAC, the jitter effect of one unit device gets divided by a factor of 1000 which is equivalent to 30-dB noise reduction at the output. It is the main advantage of the proposed *incremental* PWM method over full-PWM. As such, the jitter of the inverter chain is omitted in the simulations.

C. Comparison of Centered and Noncentered PWM

Centered and noncentered PWM show quite different implementational characteristics, so that they require different architectures. One major architectural difference is that the proposed polar transmitter employing centered PWM requires *Pulse Center* block as explained in [6]. *Pulse Center* block is crucial for centered PWM to guarantee that the pulse position is centered regardless of process variation and channel selection. It can be simply implemented with a digitally controlled delay cell array. In the transmitter with centered PWM, therefore, a separate control signal for the *Pulse Center* block is necessary

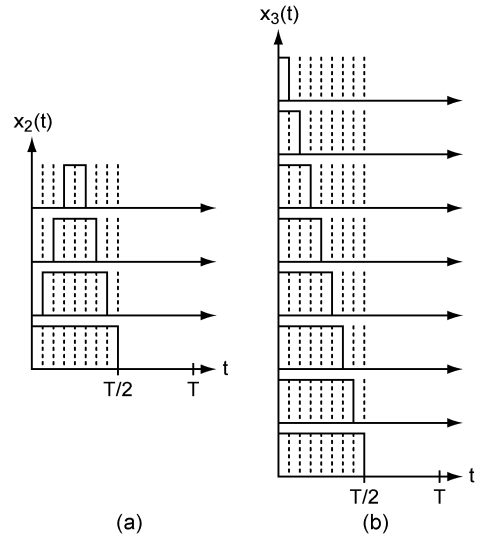


Fig. 20. The resolution comparison of (a) centered PWM and (b) noncentered PWM generators with a unit delay of $(1/8) \times (T/2)$.

to adjust the pulse position of PWM during channel selection. Calibration process is also required to compensate the process variation of delay cells in the block.

The PWM generators for each case also show different features in a practical implementation. Moreover, the resolution of each PWM generator is different even if they have the same delay chains with the same unit delay, assuming that the PWM generators shown in [6] and Fig. 14 are employed.

Fig. 20 explains the reason for the different PWM resolutions. The unit delay time is $T/16$. The number of possible centered PWM signals is only 4 because the pulse should be located at the center in all cases, but the number of noncentered PWM is 8 as shown. Consequently, the time resolution of noncentered PWM is twice the time resolution of centered PWM. As such, noncentered PWM can show better amplitude resolution improvement than centered PWM. However, it is worth mentioning that the amplitude resolution of noncentered PWM is also affected by the resolution of the phase path of the system, although the phase resolution of a digital PLL is digitally controlled and it is typically made very fine without much burden. The amplitude resolution of noncentered PWM might also reduce for a certain baseband signal, such as a signal with large peak-to-minimum ratio, unless the system employs the full size LUTs for the amplitude and phase predistortion rather than the reduced size LUTs based on approximation. Since the baseband amplitude of an EDGE transmitted signal has a limited peak-to-minimum ratio, noncentered PWM could be the better choice due to its lower complexity, thus lower noise contribution, of the RF signal path. The higher corresponding complexity of the digital predistortion is likely not to be an issue in nanometer-scale CMOS. Table V summarizes the comparison of the proposed architectures with the centered and noncentered PWM. There is a trade-off for PWM choice between the necessity of phase compensation and PWM generator's complexity.

V. BEHAVIORAL SIMULATIONS OF THE PROPOSED ARCHITECTURE WITH THE PWM SCHEME

The effect of the proposed amplitude resolution improvement using centered and noncentered PWM is verified by a behavioral

TABLE V
ARCHITECTURES WITH CENTERED PWM VERSUS NONCENTERED PWM

	Centered PWM	Non-centered PWM
Amplitude LUT	required	required
Size of amplitude LUT	$2^{N_{frac}}$ -word	$2^{N_{int}+N_{frac}}$ -word (theory) $2^{N_{frac}}$ -word (approximation)
PWM generator circuit	2 delay chains required	1 delay chain required
Resolution of PWM generator	$\frac{T}{4 \cdot delay_{unit}}$ -level	$\frac{T}{2 \cdot delay_{unit}}$ -level
Phase compensation	not required	required
Pulse positioning	pulse position control required	pulse position control not required

simulator, CppSim [10]. The simulation files and the detailed description can be found in [11]. The main objective of the proposed architecture is to improve the amplitude resolution of a DPA. Thus, only the amplitude path is modeled in detail while the phase path is modeled as ideal. The baseband input signal is the amplitude component of an EDGE signal. The output spectrum will be compared with that of the original DPA's behavioral model to show the improvement. 3GPP specification requires 100 kHz of resolution bandwidth for output RF spectrum. For fair comparison, the spectra shown in this section will be averaged periodograms with 100 kHz of frequency resolution.

In the simulation model, the input to the DPA is 10 bits, but the amplitude signal of the baseband EDGE is 15 bits. Thus, N_{int} is 10, and N_{frac} is 5. The 5-bit fractional word controls the pulsewidth of the PWM signal. A 25-stage delay chain is modeled in CppSim, and the nominal delay of each stage is 20 ps, which is the worst case delay of an inverter stage based on postlayout simulations in 65 nm CMOS.

The nonlinearity of a DPA is modeled based on the original 10-bit DPA circuit. An amplitude predistortion LUT to linearize this nonlinearity is included in the simulation models given that the measured nonlinearity of the DPA is typically known [12].

Delay mismatch effect of the delay chain is also included in the behavioral simulations. Although the Spectre Monte Carlo simulation results show about 5% of delay mismatch, we used 10% of delay mismatch for the behavioral simulations to be on the conservative side.

A. Simulation Results With Centered PWM

The amplitude predistortion LUT for centered PWM is implemented based on the simulated nominal delay of the delay elements, and its input data is 5 bits because N_{frac} is 5. The carrier frequency in the simulation is 1.0417 GHz, of which the period is 960 ps. Although this is not a legal GSM/EDGE frequency, we chose it merely for simulation convenience since 960-ps is the integer multiple of the nominal delay of the delay stages, 20-ps. The LUTs in the proposed system needs to be built based on the clock period and the delay of the delay chain, which change by process variation and channel selection. In practice, calibration process is required to measure the clock period and the nominal delay of the delay stages for the proper LUT contents. In the simulations, this calibration process could be skipped since clock period and the nominal delay of the delay stages are known.

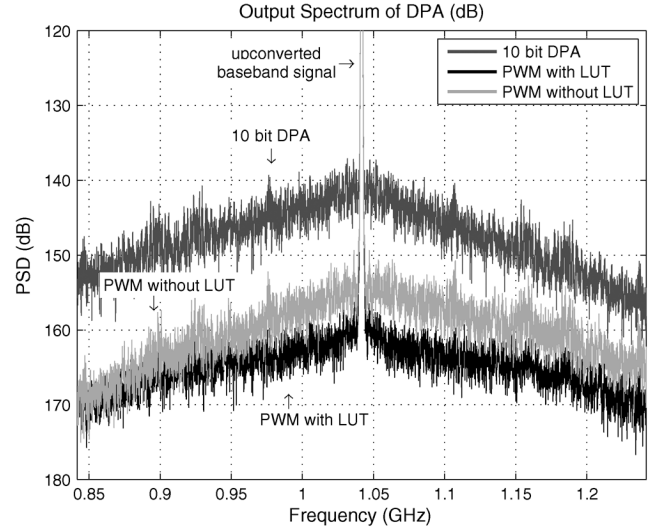


Fig. 21. Output spectrum of a 10-bit DPA and a DPA with 12-level centered PWM signals with/without the amplitude predistortion LUT.

Fig. 21 shows the spectrum of an original 10-bit DPA, and a DPA with centered PWM. It also compares the spectrum of the centered PWM with and without the amplitude predistortion LUT. The quantization noise of a DPA creates the noise skirt shown in Fig. 21. The quantization noise of the system employing 12-level PWM with the LUT is about 18 dB lower than that of the original 10-bit DPA, which means the amplitude resolution of the DPA improves by around 2.7 bits. However, the quantization noise of the centered PWM without the LUT is almost 10 dB worse than that of the PWM with the LUT.

Theoretically, the amplitude predistortion LUT for centered PWM can be built based on the exact delay of each delay stage when delay mismatch exists. However, it seems impractical since a precise method of measuring the delay would be required. In this work, we assume that the LUT for PWM is implemented based on the known nominal delay of the delay elements. In other words, the LUT is calibrated for only the process variation, and does not take care of the delay mismatch. Therefore, any delay mismatch, which is not compensated by the LUT, degrades the overall performance. Fig. 22 shows that 10% delay mismatch raises the quantization noise by up to 3 dB, which leads to 2.2-bit resolution improvement over the original 10-bit DPA. Even with the 10% delay mismatch, the spectrum of the centered PWM method satisfies the 3GPP spectral mask for the associated RX band.

Fig. 23 depicts how much amplitude resolution is degraded when the position of PWM is not exactly at the center. Up to 6 dB more quantization noise is expected when the position of the PWM signal is offset by 18 ps. In a practical implementation, the Pulse Center block will control the pulse position properly. Therefore, the pulse position error should not limit the performance.

B. Simulation Results for Noncentered PWM

Fig. 24 compares the spectrum of an original 10-bit DPA, a DPA employing noncentered PWM only with amplitude compensation, and a DPA employing noncentered PWM with both amplitude and phase compensation. The time resolution of the noncentered PWM generator used in the simulations is 20 ps,

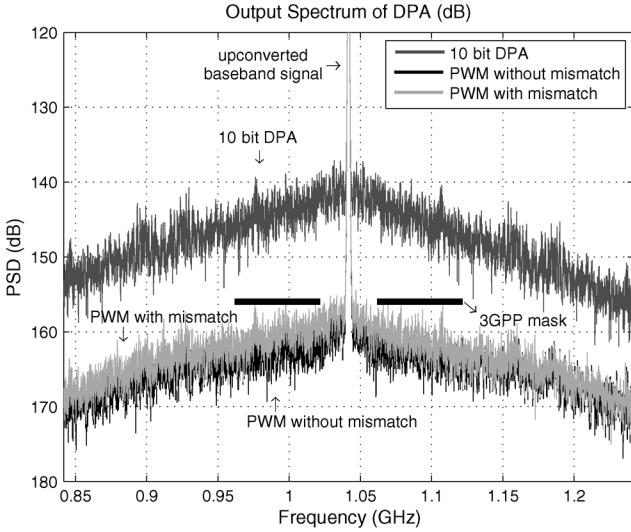


Fig. 22. Output spectrum of a 10-bit DPA and a DPA with 12-level centered PWM signals with/without delay mismatch.

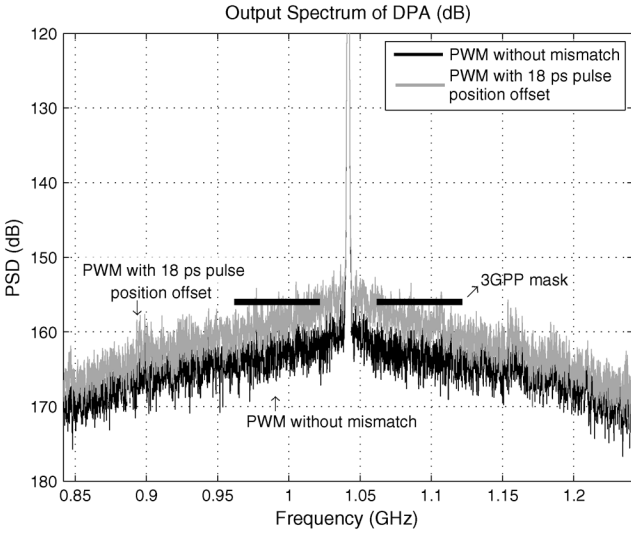


Fig. 23. Centered PWM without position offset versus centered PWM with 18-ps position offset. Delay mismatch is not considered.

which is the nominal unit delay of the delay chain circuit shown in Section IV-B. Delay mismatch is not considered. The amplitude and phase predistortion is done without the approximation technique, and the size of each LUT is 2^{15} words in this case since the baseband input data is 15 bits as was in the simulations presented in Section V-A. The quantization noise of the DPA with noncentered PWM is more than 20 dB lower than that of the original 10-bit DPA, which means the amplitude resolution is improved by at least 3 bits. Due to the higher time resolution than centered PWM, the amplitude resolution improvement is much better with noncentered PWM, but requiring an overhead of a large LUT. Fig. 24 also reveals that the quantization noise of the DPA with noncentered PWM becomes much higher if it lacks phase compensation. Thus, phase compensation is crucial for the noncentered PWM.

In the CppSim simulations, phase compensation is done by a variable delay block. The simulation time step is 10 ps, but required time resolution for phase compensation is much less than femto second for the simulation in Fig. 24. Instead of decreasing

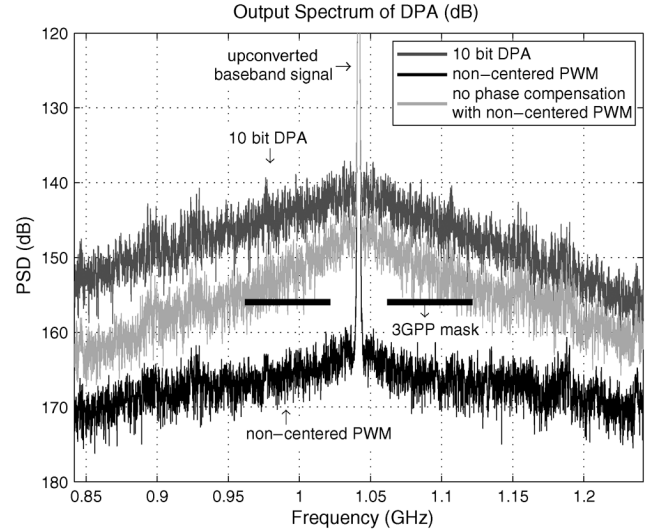


Fig. 24. Output spectrum of a 10-bit DPA, a DPA employing noncentered PWM only with amplitude compensation, and a DPA employing noncentered PWM with both amplitude and phase compensation.

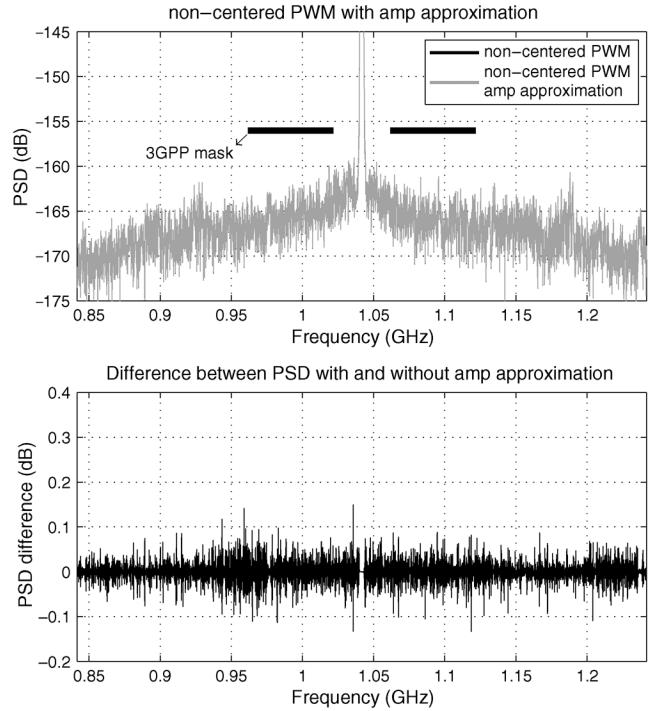


Fig. 25. Output spectrum of a DPA employing noncentered PWM with reduced size of an amplitude predistortion LUT by approximation.

the simulation time step, the signal discretization technique introduced in [13] is employed for fast simulations.

Fig. 25 shows that the amplitude predistortion with approximation for noncentered PWM, which is introduced in (8), is adequate for an EDGE signal. The difference between the spectra for noncentered PWM with ideal amplitude LUT and approximated amplitude LUT is less than ± 0.2 dB. Therefore, the reduced-size amplitude LUT with approximation can be employed for simpler implementation without resolution degradation.

Fig. 26 proves that the phase compensation with approximation is also good enough for an EDGE signal. The spectra of the ideal and approximated phase compensation are almost the

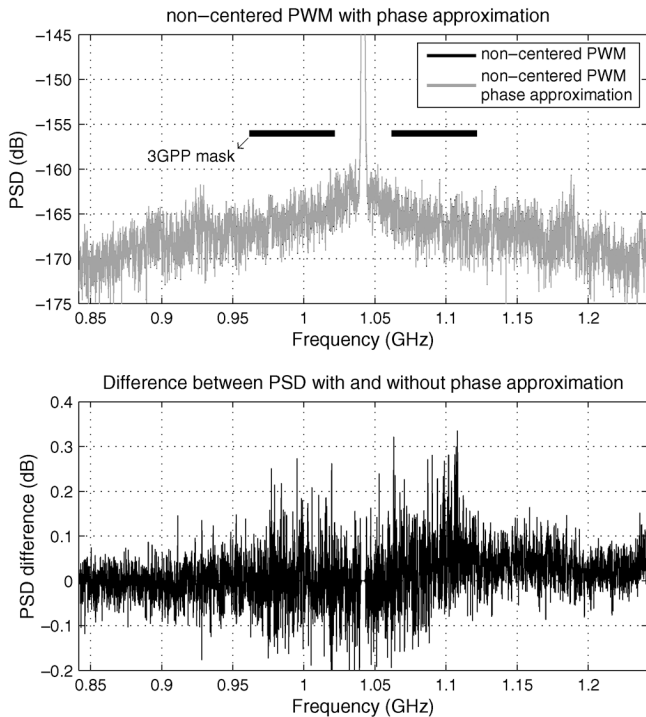


Fig. 26. Output spectrum of a DPA employing noncentered PWM with reduced size of a phase predistortion LUT by approximation.

same in Fig. 26, and the difference between the two spectra is less than ± 0.4 dB. Accordingly, the architecture with reduced size LUTs proposed in Fig. 13 is sufficient for an EDGE signal.

Even with the reduced-size amplitude and phase predistortion LUTs based on approximation, noncentered PWM shows better resolution improvement than centered PWM mainly because the time resolution of the noncentered PWM generator is twice as much as that of the centered PWM, as pointed out in Fig. 20. In practice, however, the resolution of the phase path in a digital PLL should be taken into consideration for the noncentered PWM case. Fig. 27 shows the spectra with different phase resolutions. It is obvious that the quantization noise rises if the phase resolution is low. Compared with Fig. 22, Fig. 27 shows that noncentered PWM is worse than centered PWM if the resolution of the phase path is lower than 0.05° even with the twice time resolution of the PWM generator. Therefore, the phase resolution of a PLL is another design constraint for noncentered PWM, in general. Note that the typical frequency resolution in ADPLL is 1.5 Hz [1], which will make about 100,000 times better phase resolution than 0.05° at 1 GHz of carrier frequency. Thus, the phase resolution is not an issue.

The Spectre Monte Carlo simulation results in Section IV-B are applied to the behavioral simulations to see the mismatch effect on the spectrum. Fig. 28 shows an example of the spectrum for noncentered PWM with 10% delay mismatch. The phase resolution is 0.005° in the simulations. The amplitude and phase LUT are realized based on approximation to reduce the size of the LUTs. Delay mismatch increases the quantization noise level by around 3 dB. In other words, delay mismatch lowers the resolution by around 0.5 bits. Therefore, the amplitude resolution improves by about 2.5 bits if noncentered PWM is employed with 10% delay mismatch.

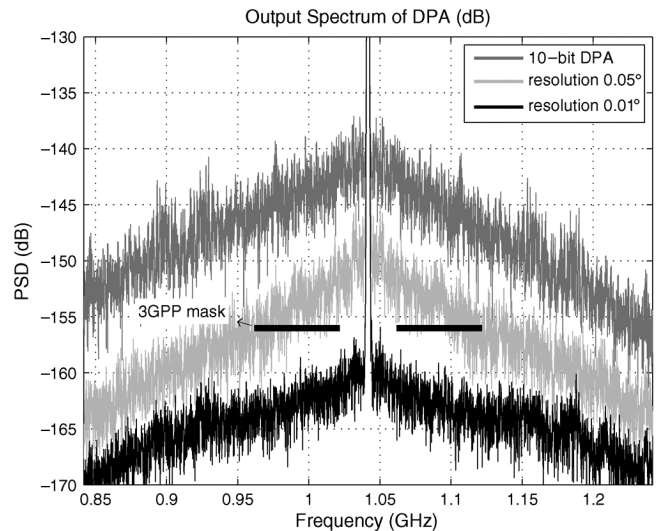


Fig. 27. Output spectrum of a DPA employing noncentered PWM with the different resolutions of a phase path.

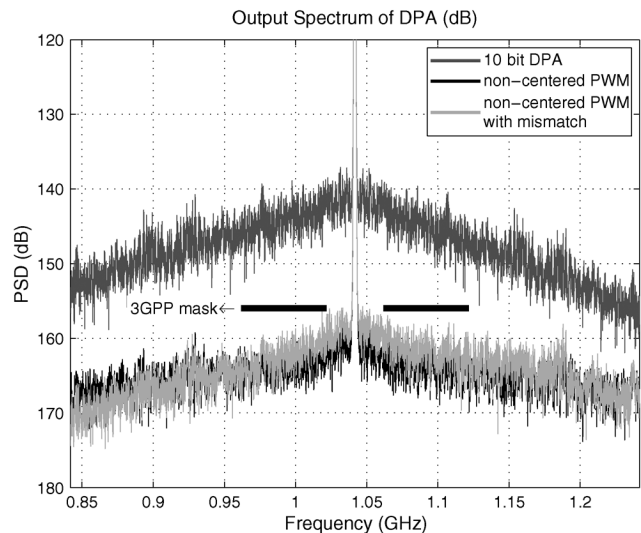


Fig. 28. The spectrum of noncentered PWM with delay mismatch.

VI. FUTURE EXTENSIONS

Centered PWM discussed in this paper has assumed that the pulse position is perfectly centered, which requires that the pulsewidth of the centered PWM be an even number of the unit delay as shown in Fig. 20. This requirement leads to centered PWM having half of the resolution of noncentered PWM. One should note that alternative approaches are possible which fit neither of these two extremes, such as allowing odd and even numbers of unit delays while striving for a *pseudocentered PWM* with alternating pulse position. Similar techniques as presented in this paper could be applied in such cases, such as phase compensation using a lookup table approach. As such, while the focus on this paper has been on centered and noncentered PWM, it is hoped that the techniques presented here will provide general intuition and direction when considering pulsewidth modulation in high performance wireless systems.

VII. CONCLUSION

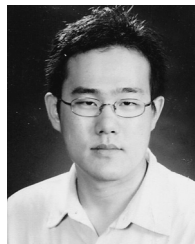
We have proposed a new architecture for amplitude resolution improvement of an RF-DAC using time as the key signal do-

main. The technique employs *incremental* PWM achieved with the fine precision of an inverter delay. Since it exploits the fine timing resolution of nanometer-scale CMOS technology, it does not require tighter device matching.

We have categorized PWM into centered and noncentered PWM, and showed that they have different characteristics from the standpoint of RF path and digital compensation complexities. The transmitter structures for both centered and noncentered PWM are proposed. They are compared in the context of the performance and implementation methods. For EDGE, the simulations with centered PWM and noncentered PWM show about 2.2-bit and 2.5-bit resolution improvement, respectively, in a 1-GHz RF-DAC generating the EDGE envelope, assuming 20-ps time granularity of delay chains and including 10% delay mismatch. The proposed architecture can be simply attached to a digitally intensive polar transmitter without major modifications. The building blocks required for the proposed architectures are a PWM generator, a fine-resolution delay controller, and amplitude/phase predistortion LUTs for PWM signals.

REFERENCES

- [1] R. Staszewski, J. Wallberg, S. Rezeq, C.-M. Hung, O. Eliezer, S. Vemulapalli, C. Fernando, K. Maggio, R. Staszewski, N. Barton, M.-C. Lee, P. Cruise, M. Entezari, K. Muhammad, and D. Leipold, "All-digital PLL and transmitter for mobile phones," *IEEE J. Solid-State Circuits*, vol. 40, no. 12, pp. 2469–2482, Dec. 2005.
- [2] J. Mehta, R. B. Staszewski, O. Eliezer, S. Rezeq, K. Waheed, M. Entezari, G. Feygin, S. Vemulapalli, V. Zoicas, C.-M. Hung, N. Barton, I. Bashir, K. Maggio, M. Frechette, M.-C. Lee, J. Wallberg, P. Cruise, and N. Yanduru, "A 0.8 mm² all-digital SAW-less polar transmitter in 65 nm EDGE SoC," in *Proc. of IEEE Solid-State Circuits Conf.*, Feb. 2010, pp. 58–59.
- [3] A. Jerng and C. G. Sodini, "A wideband $\Delta\Sigma$ digital-RF modulator for high data rate transmitters," *IEEE J. Solid-State Circuits*, vol. 42, no. 8, pp. 1710–1722, Aug. 2007.
- [4] P. Eloranta, P. Seppinen, S. Kallioinen, T. Saarela, and A. Pärssinen, "A multimode transmitter in 0.13 μm CMOS using direct-digital RF modulator," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2774–2784, Dec. 2007.
- [5] R. B. Staszewski, "Digital Deep-Submicron CMOS Frequency Synthesis for RF Wireless Applications," Ph.D. dissertation, Univ. Texas, Dallas, Aug. 2002.
- [6] M. Park, M. H. Perrott, and R. B. Staszewski, "A time-domain resolution improvement of an RF-DAC," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 7, pp. 517–521, Jul. 2010.
- [7] S. E. Meninger and M. H. Perrott, "A fractional-N frequency synthesizer architecture utilizing a mismatch compensated PFD/DAC structure for reduced quantization-induced phase noise," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 50, no. 11, pp. 839–849, Nov. 2003.
- [8] M. Nielsen and T. Larsen, "An RF pulse width modulator for switch-mode power amplification of varying envelope signals," in *Proc. Silicon Monolithic Integr. Circuits RF Syst. Top. Meet.*, Jan. 2007, pp. 277–280.
- [9] R. B. Staszewski, D. Leipold, and P. T. Balsara, "Just-in-time gain estimation of an RF digitally-controlled oscillator for digital direct frequency modulation," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 50, no. 11, pp. 887–892, Nov. 2003.
- [10] M. Perrott, Cppsim Behavior Simulator Package [Online]. Available: <http://www.cppsim.com>
- [11] M. Park, Behavioral Simulation of an Amplitude Resolution Improvement for an RF-DAC Employing PWM Scheme [Online]. Available: <http://www.cppsim.com/tutorials.html>
- [12] J. Mehta, V. Zoicas, O. Eliezer, R. B. Staszewski, S. Rezeq, M. Entezari, and P. T. Balsara, "An efficient linearization scheme for a digital polar EDGE transmitter," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 3, pp. 193–197, Mar. 2010.
- [13] M. Perrott, "Fast and accurate behavioral simulation of fractional-n frequency synthesizers and other pll/dll circuits," in *Proc. 39th Design Autom. Conf.*, 2002, pp. 498–503.



Min Park (S'05–M'08) received the B.S. degree in electronic engineering from Yonsei University, Seoul, Korea, in 1997, the M.S. degree in electrical engineering from California Institute of Technology, Pasadena, in 2003, and the Ph.D. degree in electrical engineering and computer science from Massachusetts Institute of Technology, Cambridge, in 2009.

In Summer 2008, he was a Summer Intern in the Digital RF Processor group, Texas Instruments, Dallas, TX, where he was involved with the research of EDGE RF transmitter architecture. Since 2009, he has been with Maxim Integrated Products, Sunnyvale, CA, where he designs mixed-signal circuits focusing on video amplifiers and data converters.



Michael H. Perrott (S'87–M'91–SM'09) received the B.S. degree in electrical engineering from New Mexico State University, Las Cruces, in 1988, and the M.S. and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, in 1992 and 1997, respectively.

From 1997 to 1998, he worked at Hewlett-Packard Laboratories, Palo Alto, CA, on high speed circuit techniques for Sigma-Delta synthesizers. In 1999, he was a Visiting Assistant Professor at the Hong Kong University of Science and Technology, and taught a course on the theory and implementation of frequency synthesizers. From 1999 to 2001, he worked at Silicon Laboratories, Austin, TX, and developed circuit and signal processing techniques to achieve high performance clock and data recovery circuits. He was an Assistant and then Associate Professor in Electrical Engineering and Computer Science at the Massachusetts Institute of Technology from 2001 to 2008. He was with SiTime Corporation from 2008 to 2010, where he developed key technology for MEMS-based oscillators. He is currently a Professor at Masdar Institute, Abu Dhabi, where he is focusing on low power, mixed-signal circuits for healthcare and other applications.



Robert Bogdan Staszewski (S'94–M'97–SM'05–F'09) received the B.S.E.E. (*summa cum laude*), M.S.E.E., and Ph.D. degrees from University of Texas, Dallas, in 1991, 1992, and 2002, respectively.

From 1991 to 1995 he was with Alcatel Network Systems, Richardson, TX, working on SONNET cross-connect systems for fiber optics communications. He joined Texas Instruments, Dallas, TX, in 1995 where he was elected Distinguished Member of Technical Staff. Between 1995 and 1999, he was engaged in advanced CMOS read channel development for hard disk drives. In 1999, he co-started a Digital RF Processor (DRP) group within Texas Instruments with a mission to invent new digitally intensive approaches to traditional RF functions for integrated radios in deeply scaled CMOS processes. He was appointed a CTO of the DRP group between 2007 and 2009. Since July 2009 he has been Associate Professor at Delft University of Technology, The Netherlands. He has authored and coauthored one book, three book chapters, 120 journal and conference publications, and holds 70 issued and 40 pending U.S. patents. His research interests include nanoscale CMOS architectures and circuits for frequency synthesizers, transmitters, and receivers.