Digitally-Enhanced Phase-Locking Circuits

(Invited Paper)

Pavan Kumar Hanumolu, Gu-Yeon Wei[1], Un-Ku Moon, and Kartikeya Mayaram
School of EECS, Oregon State University, Corvallis, OR 97331
[1]SEAS, Harvard University, Cambridge, MA 02138

***Abstract*— In this paper, we present an overview of digital techniques that can overcome the drawbacks of analog phase-looked loops (PLLs) implemented in deep-submicron CMOS processes. The design of key building blocks of digital PLLs such as the time-to-digital converter and digital-to-frequency converters are discussed in detail. The implementation and measured results of two digital PLL architectures, (1) based on a digitally controlled oscillator and (2) based on a digital phase accumulator, are presented. The experimental results demonstrate the feasibility of using digital PLLs in digital systems requiring high-performance PLLs.**

I. Introduction

Phase-locked loops (PLLs) are an integral component found in wireline and wireless communication systems, microprocessors, and in general, many other systems that require a clock. A vast majority of these PLLs are implemented using a *charge-pump*-based architecture [1]. Even though this analog-centric architecture continues to meet the high-performance needs of modern-day applications, PLLs implemented in deep-submicron CMOS processes incur severe cost penalties in terms of area, time-to-design, and design flexibility. In particular, the constraints imposed by continued scaling of CMOS technology such as the reduced supply voltage, increased leakage, and poor analog performance of transistors, coupled with the need to integrate sensitive analog circuits alongside a large digital core pose many design challenges.

While aggressive scaling of CMOS technology creates several obstacles and new challenges for analog designers, it continues to offer benefits for digital circuit designers. Consequently, there has been effort to exploit these benefits to implement traditionally analog functions such as phase-locking with digital building blocks. The focus of this paper is to elucidate both system- and circuit-level design tradeoffs related to implementing digital phase-locked loops. The emphasis is on developing design intuition using simplified analysis, albeit at the expense of sacrificing mathematical rigor.

The paper is organized as follows: Section II discusses the detrimental effects of scaling on analog PLLs, and the associated cost to overcome these effects. Section III provides a brief overview of digital PLLs. The analysis and design of an important building block in all digital PLLs, namely, the time-to-digital converter, is presented in Section IV. Section V, and Section VI describe the design and implementation details of two digital PLL architectures. The lock range, phase-tracking range, and frequency-tracking ranges are derived and the design tradeoffs to optimize overall performance are also discussed in these sections. Measurement results obtained from two prototype test chips are presented in Section VII and, finally, the key contributions of this paper are summarized in Section VIII.

II. Analog PLLs in Deep-Submicron CMOS

The block diagram of a commonly-used charge-pump-based analog phase-locked loop is shown in Fig. 1. The phase

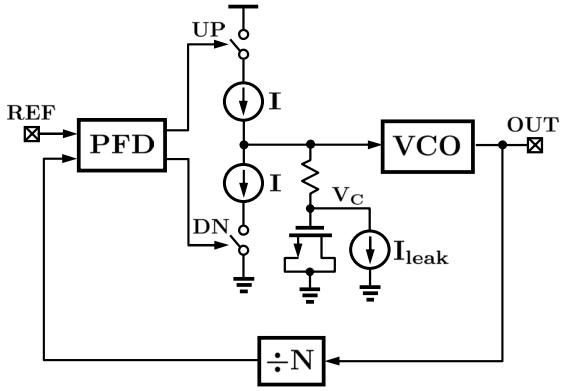

Fig. 1. Conventional analog PLL.

frequency detector (PFD) and divider are implemented using digital circuit elements and are hence less susceptible to short-channel effects. On the other hand, the charge pump, loop filter, and voltage-controlled oscillator (VCO) are severely affected by low supply voltage, low output impedance, leakage, and increased flicker noise, all associated with nanoscale CMOS technologies. To meet the small area requirements, these PLLs employ MOS capacitors to avoid the area penalty of using poly-poly or metal-insulator-metal (MIM) capacitors. However, as predicted by the International Technology Roadmap for Semiconductors (ITRS) [2] and illustrated in Fig. 2, increasing gate leakage in deep submicron CMOS processes, indicated by the parallel current source I_{leak} in Fig. 1, will hamper the continued use of MOS capacitors. The discharging of the control voltage due to capacitor leakage current and the charging of the capacitor through the negative feedback action of the PLL introduces a large ripple on the control voltage as illustrated in Fig. 3(a). This ripple on the control line leads to large deterministic jitter (DJ), also sometimes referred to as pattern jitter. For example, a PLL designed, in a 90nm CMOS process and simulated using foundry-provided leakage models, exhibits DJ in excess of

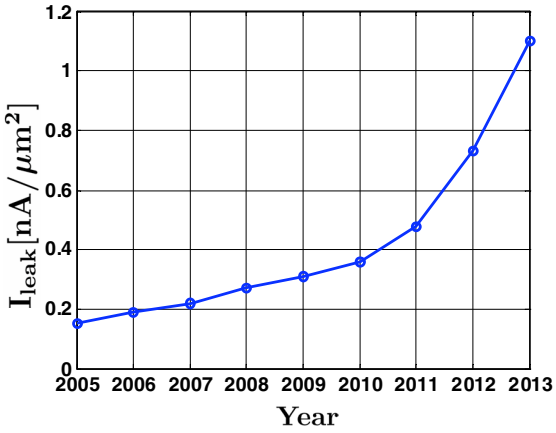

Fig. 2. ITRS gate leakage prediction.

40ps on the output of a 1GHz clock as depicted in Fig. 3(b). In this simulation, a relatively small 20pF MOS capacitor is

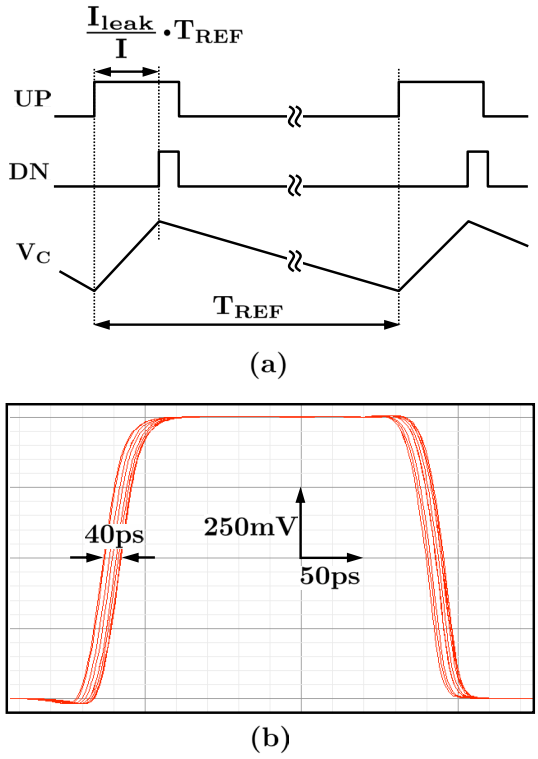

Fig. 3. PLL gate leakage issues: (a) Control voltage ripple. (b) Output clock eye diagram illustrating DJ.

used in the loop filter along with a 1pF ripple bypass capacitor (not explicitly shown in Fig. 1). It is possible to suppress leakage-induced DJ by using a MIM capacitor at the expense of increasing area by almost 20 times. However, such a large area overhead translates to a severe penalty, particularly in highly scaled CMOS processes where the chip real estate is a premium. In addition to the ripple introduced by the leakage current, current mismatch in the charge pump, exacerbated by the low supply voltage and poor transistor output impedance, also introduces ripple and further increases DJ. It is worth noting here that in the case of data recovery PLLs, the DJ caused by leakage is exacerbated by missing transitions in the incoming data.

III. Overview of Digital PLLs

A digital counterpart to the analog PLL shown in Fig. 1 can be arrived at by using a simple continuous- to discrete-time transformation. A relationship between the discrete-time operator $z = e^{j\omega T}$ and the continuous-time operator $s = j\omega$, where ω is the angular frequency of interest and T is the sampling period, can be derived using a first-order Taylor series expansion of z as shown below:

$$z = e^{j\omega T} \approx 1 + j\omega T = 1 + sT \Rightarrow s = \frac{1 - z^{-1}}{T \cdot z^{-1}}. \quad (1)$$

The above equation is valid only under the assumption that $\omega \ll 1/T$, which is true in practice since the bandwidth of most PLLs is typically much smaller than the reference frequency. We use Eq. (1) to transform the analog loop filter to a digital loop filter (DLF) as follows:

$$i_{cp}R + \frac{i_{cp}}{Cs} \Rightarrow i_{cp}R + \frac{i_{cp}T}{C}\frac{z^{-1}}{1 - z^{-1}}. \quad (2)$$

Now, using Eq. (2), we arrive at the digital PLL (DPLL) architecture shown in Fig. 4. The proportional and integral

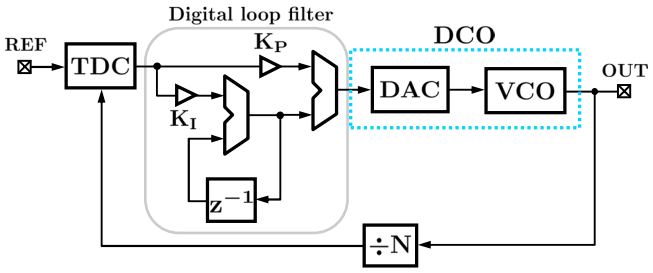

Fig. 4. A digital PLL obtained by an s-to-z transformation.

gains are given by K_P and K_I and are equal to $i_{cp}R$ and $i_{cp}T/C$, respectively. A digital-to-analog converter (DAC) interfaces the DLF to the voltage-controlled oscillator (VCO). Interested readers can refer to [3] for an alternate method, based on a bilinear-transformation of the analog loop filter, to determine the digital loop filter coefficients. In this DPLL, the time-to-digital converter (TDC) generates a digital word proportional to the phase difference between the reference input and the feedback signal.

A DPLL offers several advantages at both system and circuit levels. First, the DLF allows for a compact circuit realization and naturally eliminates the deterministic jitter caused by capacitor leakage and charge pump current mismatch. Second, since the DPLL's loop dynamics are set by DLF coefficients, loop characteristics can be easily programmed and are also immune to process, voltage, and temperature (PVT) variations. Third, the digital circuits scale gracefully to newer processes

and, as a result, mitigate the portability issues associated with analog PLLs. It has been shown in [4] that the large loop filter capacitor found in an analog PLL couples substantial amounts of substrate noise into the sensitive control voltage node. The digital LF eliminates this noise coupling problem.

Digital PLLs can be broadly classified into two categories based on the type of oscillator used: (1) a digitally controlled oscillator-based DPLL, and (2) a digital phase accumulator-based DPLL. Before discussing the design details of these two types of DPLLs, the analysis and design of various time-to-digital converters that play a crucial role in the overall performance of all DPLLs is presented.

IV. Time-to-Digital Converters

A time-to-digital converter (TDC) detects the phase difference between its inputs, and generates a digital word that is proportional to the input phase difference. The phase quantization error due to the finite resolution of the TDC is a major source of noise in DPLLs and, therefore, should be minimized to achieve good jitter performance. We will first look at TDCs for DPLLs and then show how one can extend them for digital clock and data recovery applications.

A. TDCs for Digital PLLs

The simplest implementation of a TDC is based on a D-flip flop (DFF) as shown in Fig. 5. Sample waveforms illustrating the basic operation of the DFF TDC and its transfer function are also shown in the figure. A DFF simply detects the sign of the phase error and hence serves as a 1-bit TDC. The grossly

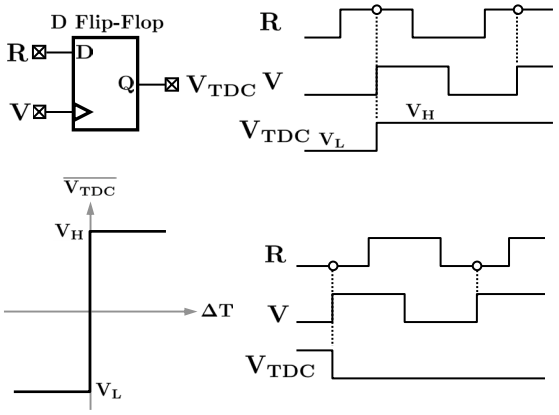

Fig. 5. D-flip flop based time-to-digital converter.

nonlinear transfer curve of the DFF TDC can be linearized in the presence of input clock jitter. Assuming a Gaussian jitter distribution, it has been shown in [5], that the linear range of TDC is extended to approximately twice the standard deviation of jitter. However, the effect of the DFF voltage offset on the TDC transfer curve was ignored in [5]. Including the voltage offset of the DFF, the average DFF TDC output $\overline{V_{TDC}}$ can be calculated to be,

$$\overline{V_{TDC}(\Delta T)} = 2V_o \cdot \mathrm{erf}\left(\frac{\Delta T - T_{OS}}{\sigma_j}\right). \quad (3)$$

where V_o is equal to $V_H - V_L$, ΔT is the input time difference, T_{OS} is the time offset resulting from the voltage offset of the DFF and is equal to $\frac{V_{os}}{t_{rise\ time}}$, and σ_j is the root mean square value of the input jitter. The small-signal gain of the TDC is given by,

$$K_{TDC} = \frac{d\overline{V_{TDC}(\Delta T)}}{d\Delta T} = \frac{2V_o}{\sqrt{2\pi}\sigma_j} e^{\left(-\frac{\Delta T}{\sqrt{2}\sigma_j}\right)^2}. \quad (4)$$

The simulated transfer characteristic of the DFF TDC with no offset and 30mV offset are shown in Fig. 6. Interestingly, the offset voltage of the TDC translates to a static phase offset in the DPLL.

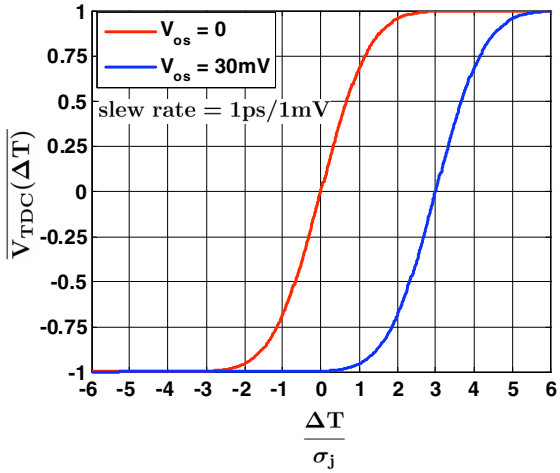

Fig. 6. DFF TDC transfer function.

Another implementation of the TDC, referred to as a stochastic TDC, exploits the inherent mismatch properties of MOS transistors in deep sub-micron processes to achieve finer phase detection resolution. A conceptual block diagram of the stochastic TDC is shown in Fig. 7 [6].

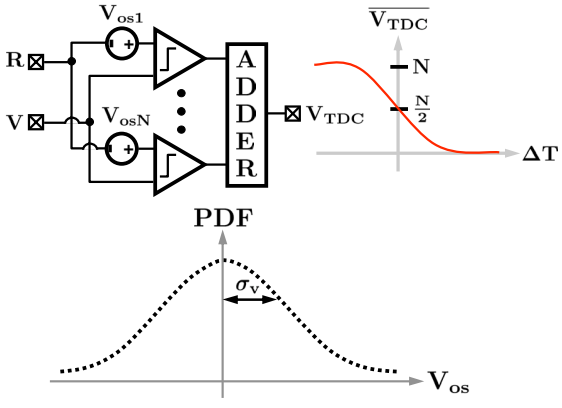

Fig. 7. Stochastic time-to-digital converter.

The inputs to the TDC are connected to a large number of nominally similar arbiters. Random transistor mismatch causes these arbiters to have different offsets. These offsets can be

modeled with a Gaussian distribution having zero mean and a standard deviation of σ_v. The random voltage offsets cause the output of arbiters to be determined by the time difference between the inputs and it can be shown that the resolution of the statistical TDC is equal to

$$\Delta\Phi = 2\pi \frac{\sqrt{2\pi}\sigma_v}{t_{\text{rise time}} \cdot N \cdot T_R}, \tag{5}$$

where $t_{\text{rise time}}$ is the input rise time and N is the number of arbiters.

$$K_{TDC}\big|_{\Delta T=0} = \frac{N \cdot t_{\text{rise time}}}{\sqrt{2\pi}\sigma_v}. \tag{6}$$

B. TDCs for Digital CDRs

The TDCs presented thus far rely on the existence of transitions every clock period and are, therefore, only suitable for DPLLs. A well-know early-late or Alexander TDC extends the DFF TDC to operate with random data [7]. In the Alexander TDC, the input random data is over-sampled by a factor of 2, and simple digital logic determines if the phase difference is positive or negative, even in the presence of missing transitions in the input data.

A major difference between the operation of the DFF TDC in DPLLs and the Alexander TDC in digital CDRs is that the latter operates on both the positive and negative edges of the input data. The former only uses either the positive or the negative edge. This subtle difference exacerbates the effects of voltage offsets and leads to a dead zone in the TDC. In order to understand this phenomenon, consider Fig. 8, which depicts the evaluation of the Alexander TDC output probability in the presence of clock jitter and voltage offset (V_{os}).

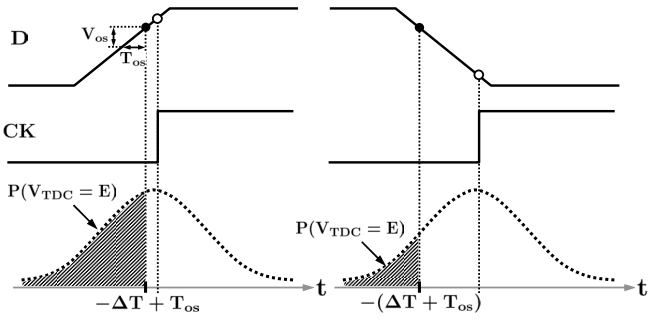

Fig. 8. Evaluation of the Alexander TDC output probability in the presence of clock jitter and voltage offset.

A simple calculation reveals that the average output of the Alexander TDC for a given input phase difference ΔT is equal to:

$$\overline{V_{PD}(\Delta T)} = V_o \cdot \text{erf}\left(\frac{\Delta T + T_{os}}{\sigma_j}\right) + V_o \cdot \text{erf}\left(\frac{\Delta T - T_{os}}{\sigma_j}\right). \tag{7}$$

Eq. (7) plotted in Fig. 9 for various voltage offsets indicates that voltage offsets in the TDC lead to a dead zone. This result was also verified by simulation in [8]. Therefore, it is important to cancel receiver offsets in clock and data recovery

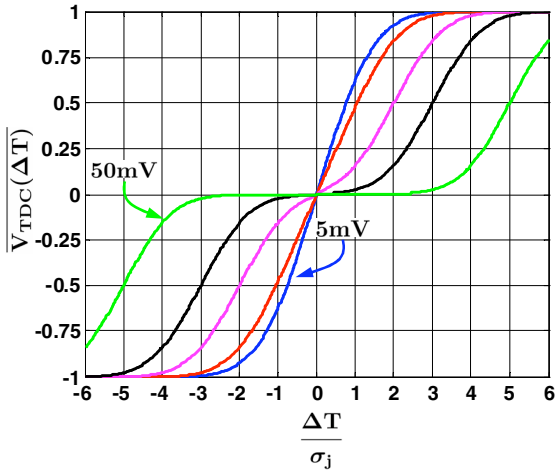

Fig. 9. Alexander TDC transfer characteristics in the presence of voltage offsets equal to 5mV, 10mV, 20mV, 30mV, and 50mV.

systems not only to improve sensitivity but also to minimize jitter resulting from the voltage-offset-induced dead zone.

It is a common practice to employ parallelism to alleviate the high speed requirements of the receiver circuitry. In these systems, multiple, evenly-spaced, lower-frequency clock phases are employed to recover high-speed data. For example, a half-rate CDR recovers data by appropriately phase-locking a four-phase clock. Random mismatch and systematic offsets cause the multiple clock phases to be non-uniformly spaced. The mismatch manifests itself as phase offsets and affect the half-rate Alexander TDC in a way similar to a voltage offset in a full-rate TDC discussed earlier. Using similar analysis and the probability distribution of the TDC output in Fig. 10, the average output of the TDC is given by,

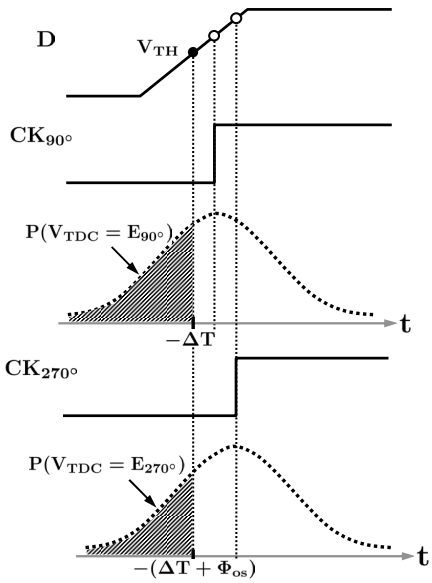

Fig. 10. Evaluation of the half-rate Alexander TDC output probability in the presence of phase offsets.

$$\overline{V_{PD}(\Delta T)} = V_o \cdot \mathrm{erf}\left(\frac{\Delta T}{\sigma_j}\right) + V_o \cdot \mathrm{erf}\left(\frac{\Delta T + \Phi_{os}}{\sigma_j}\right). \quad (8)$$

A plot of Eq. (8) (see Fig. 11) reveals that the phase offsets in a multi-phase clock recovery system can introduce a dead zone in the TDC, specifically when the phase-offsets exceed $2\sigma_j$. This analysis suggests that the jitter caused by the phase-offset-induced dead zone can dominate the jitter performance of the CDR. Therefore, it is important to minimize phase offsets along with random jitter in multi-phase oscillators to optimize overall performance of the CDR.

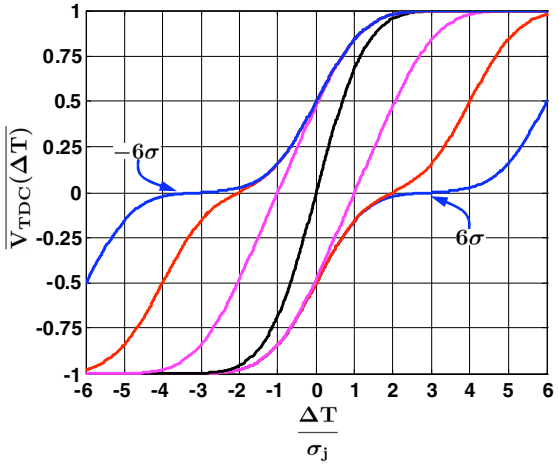

Fig. 11. Half-rate Alexander TDC transfer characteristics in the presence of phase offsets equal to -6σ_j to 6σ_j in steps of 2σ_j.

V. Digitally Controlled Oscillator-based DPLL

A digital PLL architecture, based on a digitally-controlled oscillator, was derived earlier in Section III (see Fig. 4). A digital-to-analog converter (DAC) interfaces the DLF to the analog voltage-controlled oscillator. While the current-mode DAC has been the mostly commonly used architecture [9], [10], an array of digitally controlled capacitors is employed in [11] and [12], a digitally controlled resistor is used in [13], and a bank of digitally controlled inverters is used in [14]. The implementation details of the DAC is not the focus of this paper.

A distinguishing feature of the DPLL from its analog counterpart is the nonlinearity introduced by the TDC. Recall that the nonlinearity of the 3-state phase frequency detector used in analog PLLs is only a secondary effect and does not influence the performance of the PLL in most applications. However in the case of DPLLs, nonlinearity introduce by the TDC causes the steady state condition of the DPLL to be a bounded limit cycle, which manifests itself as dither jitter on the output clock. The amplitude of this dither jitter is proportional to both the proportional step size and more importantly the loop latency. In fact, peak-to-peak dither jitter, when the proportional gain is much larger than the integral gain ($K_P \gg K_I$), can be approximated to be [15], [16],

$$DJ_{pp} \approx 2T_{latency} \cdot K_P \cdot K_{DCO}\,, \quad (9)$$

where $T_{latency}$ is the feedback loop delay and is equal to the sum of reference clock period T_{REF} and computation latency through the digital loop filter. Therefore, in order to reduce dither jitter, it is of paramount importance to minimize loop latency. Since the loop filter delay is dominated by adders, it is beneficial to split the proportional and integral paths as shown in Fig. 12 and perform the summation of the proportional and integral paths in the analog domain [17].

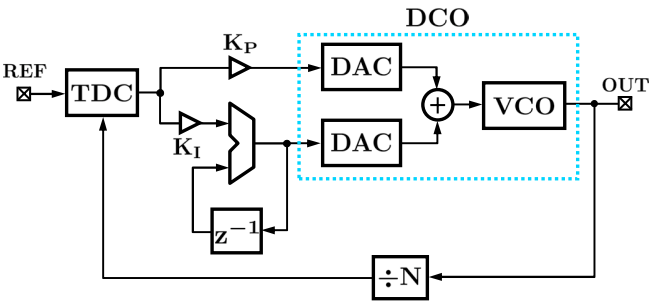

Fig. 12. DPLL with a fast proportional-path to reduce loop latency.

Having discussed the overall architecture of the DCO-based DPLL, the locking and tracking properties of this DPLL are presented next. Simple derivations illustrating the lock-in and tracking ranges along with a small-signal analysis to evaluate noise performance of the DPLL is discussed.

A. Lock Range

When the DPLL is not phase locked, and if the the frequency offset between the reference clock and the feedback clock is small, the proportional path drives the DPLL towards lock without cycle slipping. Phase lock is acquired by dithering the VCO between two frequencies ($\pm K_P \cdot K_{DCO}$) [15], resulting in an approximate lock range of:

$$\text{Lock range} \approx \pm K_P \cdot K_{DCO}. \quad (10)$$

Typically, $K_P \cdot K_{DCO}$ is chosen to achieve approximately $\pm$1500ppm of lock range. In the presence of a larger frequency error ($> K_P \cdot K_{DCO}$), the DPLL cycle slips and the integral loop drives the DCO frequency towards frequency lock in discrete steps determined by the frequency resolution of the DCO. For example, with a frequency resolution of 10ppm and an integral gain K_I equal to 2, the integral path drives the DCO frequency in steps of 20ppm.

B. Phase and Frequency Tracking

When the reference phase is modulated by a sinusoidal jitter of amplitude Φ_{IN}, the proportional path tracks the reference, if the rate of phase change due to the modulation is smaller than the proportional step size of the DCO. This results in an approximate phase tracking bandwidth given by,

$$\text{Phase tracking bandwidth} \approx \frac{K_P \cdot K_{DCO}}{2\pi\Phi_{IN}}. \quad (11)$$

This equation indicates that if the input jitter has large amplitude or if it varies with high frequency, the DPLL *slews* and, as a result, the output phase cannot track the input jitter.

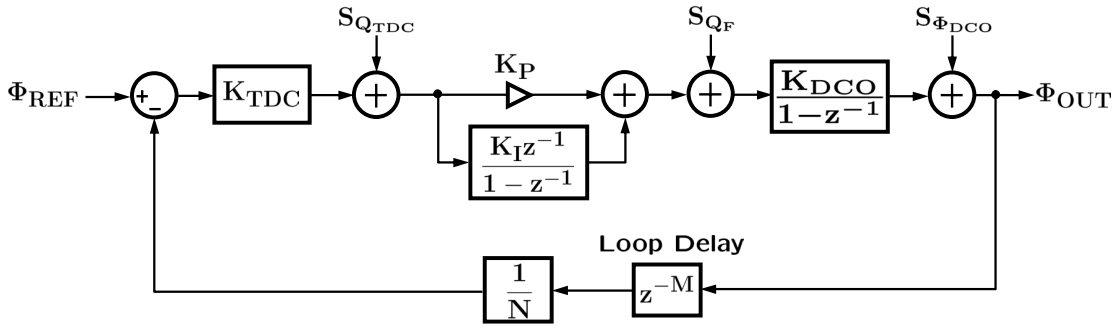

Fig. 13. Small-signal model of the DCO-based DPLL.

On the other hand, when the reference is frequency modulated with a sine wave, the integral path tracks the reference, if the rate of frequency change is slower than $\frac{\Delta F}{T_{REF}}$. This results in a frequency-tracking bandwidth given by,

$$\text{Frequency tracking bandwidth} \approx \frac{K_I \cdot K_{DCO}}{2\pi \Delta F_{IN} \cdot T_{REF}}. \quad (12)$$

If the reference frequency varies at a rate faster than $\frac{\Delta F}{T_{REF}}$, the integral loop will not be able to move the VCO frequency fast enough to track it. As a result, the integral loop slews and the phase error grows quadratically, causing the CDR to eventually lose lock.

C. Small-Signal Analysis

As shown earlier the grossly nonlinear transfer characteristic of the TDC can be linearized in the presence of clock jitter. Using the linearized gain K_{TDC} of the TDC, the small-signal model of the DPLL can be drawn as shown in Fig. 13. All of the noise sources in the DPLL, namely the TDC quantization error, DCO frequency quantization error, and DCO phase noise are represented by their respective power spectral densities $S_{Q_{TDC}}$, S_{Q_F}, and $S_{\Phi_{DCO}}$. While $S_{\Phi_{DCO}}$ is a result of intrinsic noise sources such as thermal and flicker noise, $S_{Q_{TDC}}$ and S_{Q_F} are caused by limited resolution in the circuitry. Assuming uniform distribution for the quantization error, it can be easily shown that

$$S_{Q_{TDC}} = \frac{\Delta\Phi^2}{12F_{REF}} \text{ and } S_{Q_F} = \frac{\Delta F^2}{12F_{REF}} \quad (13)$$

where $\Delta\Phi$ (see Eq. (5)) and ΔF are the resolution of the TDC and the DCO, respectively, and F_{REF} is the DPLL reference frequency.

The loop gain of the DPLL is

$$LG(z^{-1}) = \frac{K_{TDC}K_{DCO}}{1-z^{-1}}\left(K_P + \frac{K_I z^{-1}}{1-z^{-1}}\right)\frac{z^{-M}}{N}. \quad (14)$$

Analogous to the noise analysis performed for analog PLLs, the impact of each of the noise sources on the output clock jitter can be evaluated using a simple transfer function analysis. The total output phase noise $S_{\Phi_{OUT}}$, calculated using such an analysis, is given by

$$S_{\Phi_{OUT}} = \left|\frac{N}{K_{TDC}}\frac{LG(z^{-1})}{1+LG(z^{-1})}\right|^2 S_{Q_{TDC}} + \left|\frac{\frac{K_{DCO}}{1-z^{-1}}}{1+LG(z^{-1})}\right|^2 S_{Q_F} + \left|\frac{1}{1+LG(z^{-1})}\right|^2 S_{Q_{DCO}} \quad (15)$$

It is important to remember that the above equation does not account for dither jitter induced by TDC nonlinearity and loop latency.

VI. Digital Phase-Accumulator-based DPLL

An alternate method for implementing a DPLL is to use a digital phase accumulator (DPA) in place of the DCO. Analogous to the DCO, the DPA implements the digital-to-frequency function by explicitly accumulating phase in an unlimited fashion. The rate of output phase change is controlled by the input control word and, therefore, determines the DPA's frequency resolution. The simplest implementation of a DPA is to use an oscillator and a programmable digital counter as shown in Fig. 14 [18]. In this DPA, if the input

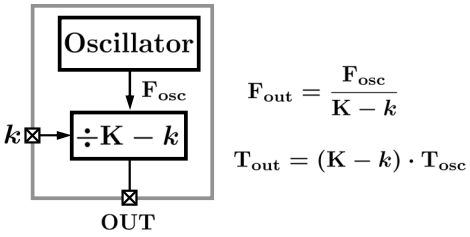

Fig. 14. Counter-based DPA.

digital control signal $k = 0$, then the output period is equal to KT_{osc}, where T_{osc} is the time period of the fixed oscillator. When k takes on a non-zero integer value, the output period equals $KT_{osc} - kT_{osc}$, thereby providing the needed frequency control, through an additional phase shift of kT_{osc} every clock period of the oscillator. The difference equation describing the DPA is given by

$$P_{OUT}[n] = P_{OUT}[n-1] + k[n-1] \cdot T_{osc}. \quad (16)$$

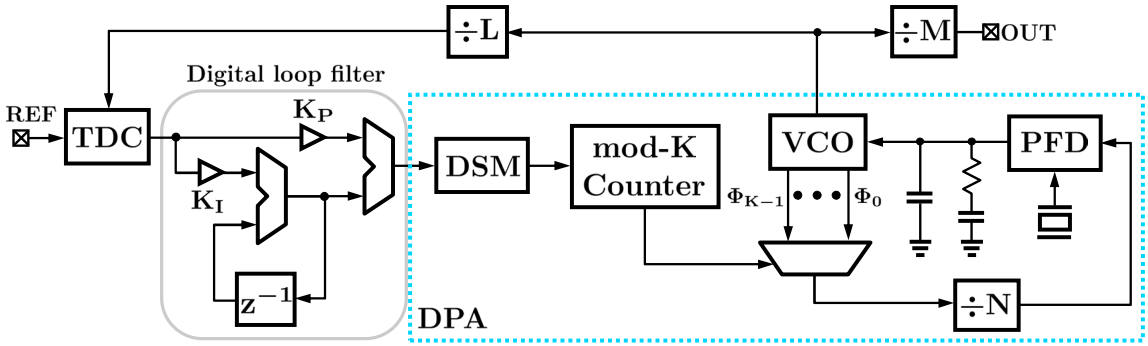

Fig. 15. DPLL employing a feed-back phase-selection DPA.

where P_{OUT} is the output time period. Taking the z-transform on both sides of this equation reveals the transfer function of DPA to be

$$\frac{P_{OUT}(z^{-1})}{k(z^{-1})} = \frac{K_\Phi z^{-1}}{1 - z^{-1}}, \tag{17}$$

where K_Φ is the DPA gain and is equal to T_{osc} for the counter-based DPA. As expected, the transfer function of the DPA is the same as the DCO except for a difference in the transfer gain.

In the counter-based DPA, as is evident from Eq. (16), the phase increments are in steps of T_{osc}, thereby mandating a very high frequency oscillator to reduce the jitter introduced by the quantization error of the DPA. The requirement of an oscillator with high frequency and good frequency stability limits the usefulness of this architecture.

In order to obviate a high-frequency oscillator, an analog PLL can be used to generate multiple phases with excellent frequency stability using only a low frequency crystal oscillator [19]. The closely spaced phases generated by the PLL reduce the jitter due to the quantization error, effectively improving the resolution of the DPA.

At this point, it might appear contradictory to use an analog PLL to overcome the difficulties of building a digital PLL. In order to understand the benefits of this approach, consider the design of a conventional analog PLL used to generate a 100MHz clock using a very low-frequency 100kHz reference signal. One application of such a PLL is to generate a *pixel clock* from the low-frequency *Hsync signal* in display drivers. There are two major difficulties in implementing such a PLL: (1) large VCO noise due to the low PLL bandwidth ($<$ 10kHz) and (2) large silicon area to implement the low-frequency loop filter. The DPA based DPLL overcomes these two drawbacks by suppressing the VCO phase noise with the large bandwidth of the analog PLL and by employing a digital loop filter to implement the low-frequency filter, thereby reducing silicon area.

Even though the phase-selecting DPA-based DPLL improves on the counter-based DPLL, jitter due to coarse phase spacing of the VCO can be prohibitively high for many applications. For instance, the deterministic jitter of a 1GHz DPA implemented with a 4-stage differential ring VCO is at least $\pm$125ps. An improved DPA-based DPLL that exploits the low-pass nature of the analog PLL to achieve excellent jitter performance is discussed next.

The detailed block diagram of the feedback phase selection DPA based DPLL is shown in Fig. 15 [20], [21]. In this architecture, the phase-selecting multiplexer is placed inside the PLL feedback loop and the voltage controlled oscillator (VCO) output serves as the feedback clock to the DPLL. The main benefit of this architecture is that the phase jumps and the phase quantization error at the output of the multiplexer are suppressed by the low-pass loop filter of the PLL. In order to further reduce phase quantization error, a delta-sigma modulator (DSM) is used to shape the quantization error to high frequencies, which is eventually filtered by the PLL. Jitter performance that is comparable to that of analog PLLs can be achieved using this architecture.

Observing that the DPA behaves much like a DCO with a gain of $\frac{\Delta\Phi_{dpa}}{T_{REF}}$, where $\Delta\Phi$ is equal to the minimum possible phase increment and T_{REF} is the update period of the DPA, the lock-in and tracking ranges of the DPA-based DPLL can be calculated in a manner similar to that of the DCO-based DPLL (see Sections V-A and V-B).

VII. Measurement Results

Two prototype chips, one implementing a DCO-based DPLL and the other implementing a DPA-based quarter-rate DPLL were fabricated in 0.13μm and 0.18μm CMOS technologies, respectively. The die-photos of these test chips are shown in Fig. 16. The active die areas of the DCO-based DPLL and DPA-based DPLL are 0.1mm^2 and 0.8mm^2, respectively. The DCO-based DPLL is based on the architecture shown in Fig. 12 [17] and the DPA-based DPLL uses the architecture shown in Fig. 15 [21]. Both of the prototype chips employ Alexander TDCs and are, therefore, capable of operating with random input data. Voltage offsets in the TDC are minimized by choosing large device sizes and symmetric layout techniques. In the case of the quarter-rate DPA-based DPLL, phase offsets were also minimized using similar design techniques. There are no means to measure the dead zone in the test chips resulting from residual voltage and phase offsets.

The measured jitter for these DPLLs, both operating at 1.6GHz, is depicted in Fig. 17. In the DCO-based DPLL, the

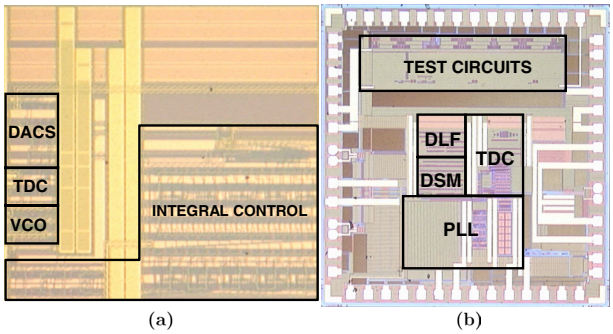

Fig. 16. Die-photos: (a) DCO-based DPLL. (b)DPA-based DPLL.

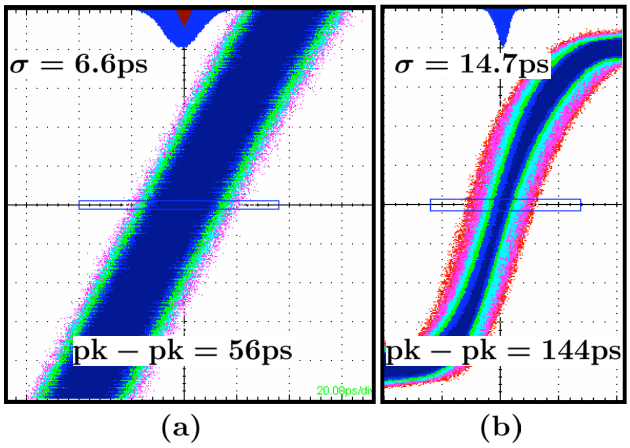

Fig. 17. Measured jitter at 1.6GHz: (a) DCO-based DPLL, (b) DPA-based DPLL.

relatively large proportional step size ($K_P \cdot K_{DCO}$) dominated the jitter. In the DPA-based DPLL, the larger than expected analog PLL bandwidth caused the unfiltered noise of the delta-sigma modulator to dominate the output jitter. The power consumption of the DCO-based DPLL is 12mW while that of the DPA-based DPLL is 18mW.

VIII. Summary

The difficulties associated with implementing analog PLLs in deep sub-micron CMOS processes are summarized and the motivation to overcome these difficulties using digital PLLs is presented. Time-to-digital converters, which perform the essential function of converting phase error to a digital output have been analyzed in detail. TDCs employed in digital PLLs and digital CDRs lead to static phase offsets and dead-zones in the TDC transfer curve, respectively. Phase mismatch in multi-phase digital CDRs also has similar effects on the TDC transfer characteristic.

Two digital PLL architectures, one based on a digitally-controlled oscillator and the other based on a digital phase accumulator, have been presented. The expressions for locking and tracking ranges of both types of PLLs are derived, and noise analysis using approximate small-signal models are provided. Finally, measured results from two test chips demonstrate the feasibility of using digital PLLs in digital systems requiring high-performance PLLs.

IX. Acknowledgements

We thank Samsung Electronics and National Semiconductor for providing IC fabrication. This work was supported by SRC under contracts 2003-HJ-1076 and 2007-HJ-1597 and, partly by Intel Corporation.

References

[1] F. Gardner, "Charge-pump phase-lock loops," *IEEE Trans. Commun.*, pp. 1849–1858, Nov. 1980.
[2] *International Technology Roadmap for Semiconductors*, 2005 Edition.
[3] V. Kratyuk, P. Hanumolu, K. Mayaram, and U. Moon, "A design procedure for all-digital phase-locked loops based on a charge-pump phase-locked-loop analogy," *IEEE Trans. Circuits Syst. II*, vol. 54, pp. 247–251, Mar. 2007.
[4] P. Larsson, "Measurements and analysis of PLL jitter caused by digital switching noise," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1113–1119, July 2001.
[5] J. Lee, K. Kundert, and B. Razavi, "Analysis and modeling of bang-bang clock and data recovery circuits," *IEEE J. Solid-State Circuits*, vol. 39, pp. 1571–1580, Sept. 2004.
[6] V. Kratyuk, P. Hanumolu, K. Ok, K. Mayaram, and U. Moon, "A digital PLL with a stochastic time-to-digital converter," in *IEEE VLSI Circuits Sym. Tech. Papers*, June 2006, pp. 31–32.
[7] J. Alexander, "Clock recovery from random binary signals," *Electr. Lett.*, pp. 541–542, Oct. 1975.
[8] J. Sonntag and J. Stonick, "A digital clock and data recovery architecture for multi-gigabit/s binary links," in *Proc. of IEEE CICC*, Sept. 2005, pp. 532–539.
[9] I. Hwang, S. Song, and S. Kim, "A digitally controlled phase-locked loop with a digital phase-frequency detector for fast acquisition," *IEEE J. Solid-State Circuits*, vol. 36, pp. 1574–1581, Oct. 2001.
[10] A. Fahim, "A compact, low-power low-jitter digital PLL," in *Proc. of IEEE ESSCIRC*, Sept. 2003, pp. 101–104.
[11] R. Staszewski *et al.*, "All-digital TX frequency synthesizer and discrete-time receiver for bluetooth radio in 130-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 39, pp. 2278–2291, Dec. 2004.
[12] C. Hung, R. Staszewski, N. Barton, M. Lee, and D. Leipold, "A digitally controlled oscillator system for SAW-less transmitters in cellular handsets," *IEEE J. Solid-State Circuits*, vol. 41, pp. 1160–1170, May 2006.
[13] D. Oh *et al.*, "A 2.8Gb/s all-digital CDR with a 10b monotonic DCO," in *ISSCC Dig. Tech. Papers*, Feb. 2007, pp. 222–223.
[14] A. Rylyakov, J. Tierno, G. English, D. Friedman, and M. Meghelli, "A wide power-supply range (0.5V-to-1.3V) wide tuning range (500 MHz-to-8 GHz) all-static CMOS AD PLL in 65nm SOI," in *ISSCC Dig. Tech. Papers*, Feb. 2007, pp. 172–173.
[15] R. Walker, C. Stout, J. Wu, B. Lai, C. Yen, T. Hornak, and P. Petruno, "Two-chip 1.5-GBd serial link interface," *IEEE J. Solid-State Circuits*, vol. 27, pp. 1806–1811, Dec. 1992.
[16] N. DaDalt, "A design-oriented study of the nonlinear dynamics of digital bang-bang plls," *IEEE Trans. Circuits Syst. I*, vol. 52, pp. 21–31, Jan. 2005.
[17] P. Hanumolu, M. Kim, G. Wei, and U. Moon, "A 1.6Gbps digital clock and data recovery circuit," in *Proc. of IEEE CICC*, Sept. 2006, pp. 603–606.
[18] G. Gill and S. Gupta, "First-order discrete phase-locked loop with applications to demodulation of angle-modulated carrier," *IEEE Trans. Commun.*, pp. 452–462, June 1972.
[19] J. Sonntag and R. Leonowich, "A monolithic CMOS 10MHz DPLL for burst-mode data retiming," in *ISSCC Dig. Tech. Papers*, Feb. 1990, pp. 194–195.
[20] H. Lee, O. Kim, K. Jung, and D. J. J. Shin, "A PVT-Tolerant low-1/f noise dual-loop hybrid PLL in 0.18μm," in *ISSCC Dig. Tech. Papers*, Feb. 2006, pp. 586–587.
[21] P. Hanumolu, G. Wei, and U. Moon, "A wide tracking range 0.2–4Gbps clock and data recovery circuit," in *IEEE VLSI Circuits Sym. Tech. Papers*, June 2006, pp. 88–89.