A 3.5 GHz Digital Fractional-*N* PLL Frequency Synthesizer Based on Ring Oscillator Frequency-to-Digital Conversion

Colin Weltin-Wu, *Member, IEEE*, Guobi Zhao, and Ian Galton, *Fellow, IEEE*

***Abstract*—A 3.5 GHz digital fractional-*N* PLL in 65 nm CMOS technology is presented that achieves phase noise and spurious tone performance comparable to those of a high-performance analog PLL. It is enabled by a new second-order frequency-to-digital converter that uses a dual-mode ring oscillator and digital logic instead of a charge pump and ADC. It also incorporates a new technique to reduce excess phase noise that would otherwise be caused by component mismatches when the DCO input is near integer boundaries. The PLL's largest in-band fractional spur is -60 dBc, its worst-case reference spur is -81 dBc, and its phase noise is -93, -126, and -151 dBc/Hz at offsets of 100 kHz, 1 MHz, and 20 MHz, respectively. Its active area is 0.34 mm² and it dissipates 15.6 mW from a 1 V supply.**

***Index Terms*—Digital PLL, fractional-*N* phase-locked loop, frequency synthesizer, frequency-to-digital conversion, PLL.**

I. Introduction

EVOLVING wireless communication standards place increasingly stringent performance requirements on the frequency synthesizers that generate RF local oscillator signals for up and down conversion in wireless transceivers. Conventional analog fractional-*N* phase-locked loops (PLLs) with digital $\Delta\Sigma$ modulation are the standard for such frequency synthesizers because of their excellent phase noise and spurious tone performance [1]–[5]. Unfortunately, they require high-performance analog charge pumps and large-area analog filters, so the trends of CMOS technology scaling and increasingly dense system-on-chip integration have created an inhospitable environment for them.

Digital fractional-*N* PLLs have been developed over the last decade to address this problem [6]–[36]. They avoid large analog loop filters and can tolerate device leakage and low supply voltages which makes them better-suited to highly-scaled CMOS technology than analog PLLs. They are increasingly used in place of analog PLLs as frequency synthesizers, but they have yet to fully replace analog PLLs in high-performance wireless applications. While both analog and digital fractional-*N* PLLs introduce quantization noise, in prior digital PLLs the quantization noise has higher power or higher spurious tones than in comparable analog PLLs. Consequently, they exhibit worse phase noise or spurious tone performance than the best analog PLLs [37]–[45]. Digital PLLs based on second-order $\Delta\Sigma$ frequency-to-digital conversion (FDC-PLLs) offer a potential solution to this problem in that their quantization noise ideally is equivalent to that of an analog PLL with second-order $\Delta\Sigma$ modulation, but prior second-order FDC-PLLs incorporate charge pumps and ADCs which have so far limited their performance and minimum supply voltages [46]–[48].

This paper presents an FDC-PLL that avoids these limitations by implementing the functionality of a charge pump and ADC with a dual-mode ring oscillator (DMRO) and digital logic [49]. The new architecture and circuit-level techniques described in the paper enable very low phase noise and low power dissipation with better spurious tone performance than previously published PLLs in the same class.

II. High-Level Architecture and Functionality

A high-level diagram of the implemented second-order FDC-PLL is shown in Fig. 1 [49]. Its input, $v_{\rm ref}(t)$, is a periodic reference signal of frequency $f_{\rm ref}$, and its output, $v_{\rm out}(t)$, ideally is periodic with frequency $(N+\alpha)f_{\rm ref}$, where N is a positive integer and α is a fractional value in the range $-1/2$ to 1/2. Its principal sub-blocks are a $\Delta\Sigma$ frequency-to-digital converter (FDC), a digitally controlled oscillator (DCO), and a lowpass digital loop filter (DLF). The main high-level architecture differences between the FDC-PLL and prior second-order FDC-PLLs lie in the $\Delta\Sigma$ FDC. In the remainder of this section, the FDC-PLL is first explained in terms of the input-output behavior of the $\Delta\Sigma$ FDC, and then the $\Delta\Sigma$ FDC's internal operation is explained.

The $y[n]$ output of the $\Delta\Sigma$ FDC is an integer-valued, $f_{\rm ref}$-rate digital sequence. It can be written as $y[n] = -\alpha - e_{\rm PLL}[n] + e_{\Delta\Sigma}[n]$, where $e_{\rm PLL}[n]$ is an estimate of the PLL's average frequency error over the nth reference period (the term *estimate* is used because $e_{\rm PLL}[n]$ also contains any noise and distortion resulting from non-ideal circuit behavior in the $\Delta\Sigma$ FDC), and $e_{\Delta\Sigma}[n]$ is quantization noise. As proven mathematically for a slightly generalized version of the $\Delta\Sigma$ FDC in [50] and explained qualitatively below, $e_{\Delta\Sigma}[n]$ is identical to the quanti-

Manuscript received April 28, 2015; revised July 13, 2015; accepted August 02, 2015. Date of publication October 30, 2015; date of current version November 24, 2015. This paper was approved by Guest Editor Jussi Ryynanen. This work was supported by the National Science Foundation under Award 1343389, by funding from Analog Devices, and by IC fabrication provided by STMicroelectronics via their membership in the UCSD Center for Wireless Communications.

C. Weltin-Wu is with Analog Devices, San Jose, CA 95134 USA.

G. Zhao is with Spreadtrum, Shanghai 201203, China.

I. Galton is with the Department of Electrical Engineering, University of California, San Diego, CA 92093 USA.

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

Digital Object Identifier 10.1109/JSSC.2015.2468712

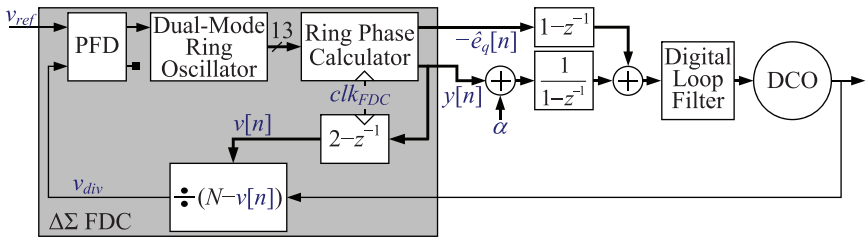

Fig. 1. High-level diagram of the implemented FDC-PLL.

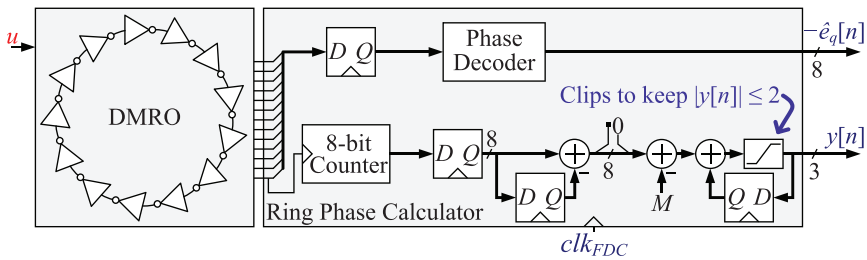

Fig. 2. Signal processing performed by the ring phase calculator.

zation noise from a second-order ΔΣ modulator. This implies that $e_{\Delta\Sigma}[n] = e_q[n] - 2e_q[n-1] + e_q[n-2]$, where $e_q[n]$ is uniform quantization noise [51]. As indicated in Fig. 1, α is added to $y[n]$ and the result is accumulated to generate an estimate of the PLL's phase error plus first-order quantization noise given by $e_q[n] - e_q[n-1]$. Therefore, each output sample of the accumulator prior to the DLF is nearly proportional to the amount of charge in each charge pump pulse in an analog PLL with second-order ΔΣ modulation [52].

As described below, the $-\hat{e}_q[n]$ output of the ΔΣ FDC is a digitized version of $-e_q[n]$ with a quantization step-size of 1/26, so it can be written as $-\hat{e}_q[n] = -e_q[n] + e_{Rq}[n]$, where $e_{Rq}[n]$ is quantization noise with much (about 28 dB) lower power than $e_q[n]$. As indicated in Fig. 1, its first difference (i.e., $-\hat{e}_q[n] + \hat{e}_q[n-1]$) is added to the output of the accumulator prior to the DLF. This has the effect of cancelling most of the quantization noise prior to the DLF, because it replaces $e_q[n] - e_q[n-1]$ with $e_{Rq}[n] - e_{Rq}[n-1]$. This quantization noise cancellation (QNC) technique is a variant of similar techniques that have been applied previously to both analog and digital PLLs [38], [39], [44], [45], [52], [53].

The DLF contains a proportional-integral compensator that controls the PLL's dynamics, and it also contains IIR filter stages that introduce four out-of-band poles to further suppress the residual ΔΣ quantization noise. The linearized model presented in [50] was used to choose the placement of the poles and zeros for this design.

ΔΣ FDC Signal Processing: The ΔΣ FDC consists of a phase-frequency detector (PFD) and multi-modulus divider of the types used in analog PLLs, a DMRO, a ring phase calculator digital block, and a $2 - z^{-1}$ digital block. The DMRO frequency switches from f_{low} to f_{high} when the PFD output, $u(t)$, goes high, and from f_{high} to f_{low} when $u(t)$ goes low, where $f_{\text{high}} - f_{\text{low}} \cong f_{\text{PLL}}$.[1]

[1]As explained in [50], the PLL's performance is not highly sensitive to deviations of $f_{\text{high}} - f_{\text{low}}$ from f_{PLL} or from non-ideal frequency switching transients.

As shown in Fig. 2, the ring phase calculator samples the outputs of the DMRO's 13 inverters to generate $-\hat{e}_q[n]$ and samples the output of an 8 bit counter clocked by one of the DMRO's inverters to generate $y[n]$. These sampling operations are synchronous with each rising edge of the f_{ref}-rate clock, clk_{FDC}, which is asynchronous with the rising edges of the DMRO inverter outputs. Synchronization circuitry that enables the sampling to occur without errors or metastability issues is described in Section III, but is omitted from Fig. 2 to simplify the ring phase calculator's signal processing explanation.

The ring phase calculator's 8 bit counter is never reset, so it counts DMRO cycles and rolls over every 256 DMRO cycles. Thus, its output, $c(t)$, can be viewed as the measured DMRO phase in cycles quantized down to the nearest integer modulo 256. The phase decoder uses all 13 DMRO inverter outputs to measure the counter's quantization error to a resolution of 1/26 of a DMRO cycle, so its output can be viewed as a quantized version of the counter's quantization error. This is illustrated in Fig. 3 wherein the sampling operation prior to the phase decoder is omitted for illustration clarity.

During the nth clk_{FDC} period, the ring phase calculator calculates the first difference of the sampled counter output (i.e., the current minus the previous sampled counter outputs) and clears the most significant bit (MSB) of the result. Frequency is the derivative of phase, so it can be verified that these operations result in a measurement of the DMRO frequency divided by f_{ref} and quantized to the nearest integer. The ring phase calculator subtracts an integer, M, from this frequency measurement and accumulates the result to generate $y[n]$ (in the prototype IC, M can be set to any integer from 40 to 80). Given that the $y[n]$ path in the ring phase calculator implements a first difference and subsequently an accumulator, $y[n]$ is a measurement of the DMRO's phase. Specifically, it can be verified that $y[n]$ is the difference between the DMRO's phase in cycles quantized down to the nearest integer and the phase of an ideal oscillator of frequency $M f_{\text{ref}}$ at the time of the nth rising edge of clk_{FDC} [50].

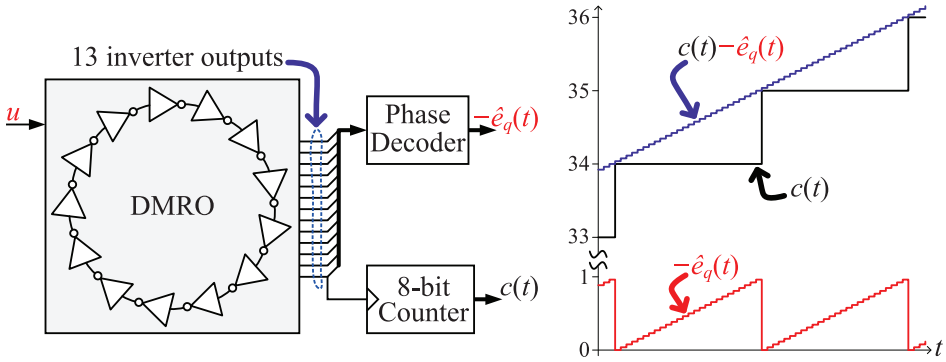

Fig. 3. Relationship between the ring phase calculator's phase decoder and 8 bit counter.

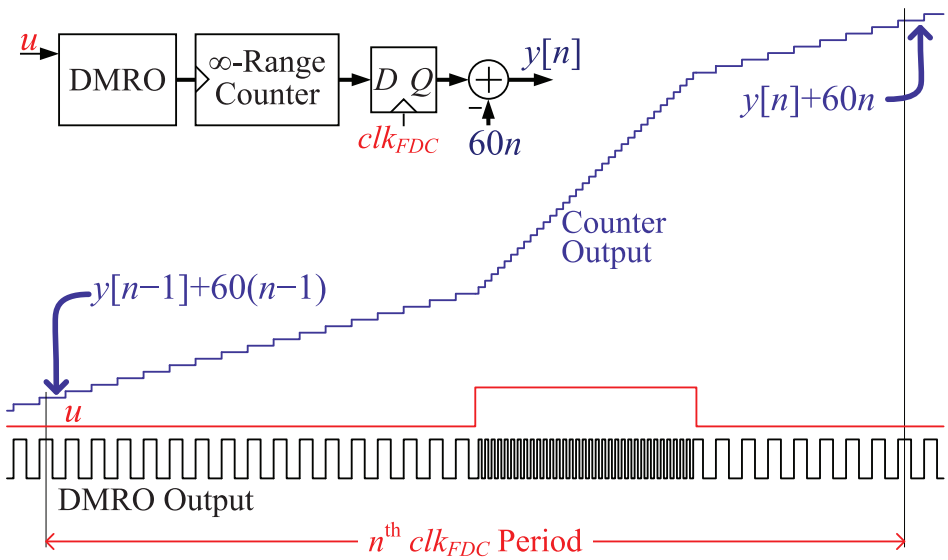

Fig. 4. Equivalent behavior of the ring phase calculator in terms of how it generates $y[n]$.

The clipper shown in Fig. 2 only affects the locking behavior of the PLL because the magnitude of $y[n]$ would be bounded by 2 when the PLL is locked even without the clipper [50]. Its purpose is to reduce the PLL's worst-case locking time. During startup, if the reference and divider were only slightly out of lock, allowing the magnitude of $y[n]$ to exceed 2 would sometimes speed up the locking process. However, when the PLL loop is far out of lock, large-magnitude $y[n]$ values can significantly increase the lock time in certain situations. The clipper circumvents this issue.

Qualitative Explanation of the ΔΣ FDC'S Operation: It can be verified from the explanation above that $y[n]$ is equivalent to the result of counting DMRO cycles with an infinite-range counter (i.e., a counter that never rolls over), sampling the counter on each rising edge clk_{FDC}, and subtracting Mn from the result. This is shown in Fig. 4 for $M = 60$ along with a corresponding timing diagram. The counter increases with a slope that is low when the DMRO frequency is f_{low} and a slope that is higher when the DMRO frequency is f_{high}. Therefore, the counter output is equivalent to the quantized integral of a constant plus the PFD output, $u(t)$, as illustrated in Fig. 4.

As shown in Fig. 1, the ΔΣ FDC has local feedback through the divider. For example, increasing the $u(t)$ pulse width during the nth clk_{FDC} period has the effect of increasing $y[n]$ (as can be seen from the timing diagram in Fig. 4), which decreases the divider modulus (because the divider modulus is $N - 2y[n] + y[n-1]$), which decreases the $u(t)$ pulse width during the $(n+1)$th clk_{FDC} period (because $u(t)$ goes low when the divider output goes high). As proven in [50], this negative feedback ensures that $u(t)$ has one pulse per clk_{FDC} period with an average duration of

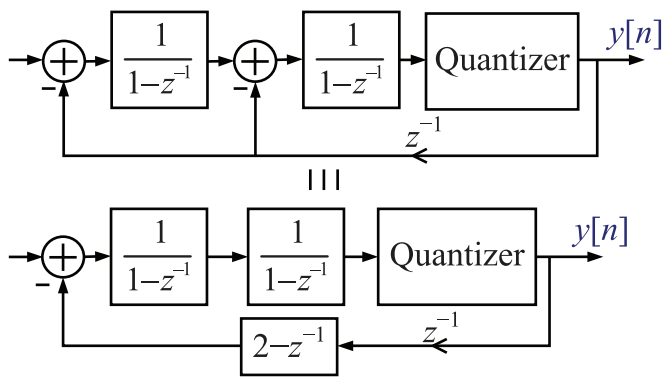

Fig. 5. Equivalent forms of a second-order ΔΣ modulator.

$$T_{\bar{u}} = \frac{M - T_{\text{ref}} f_{\text{low}}}{f_{\text{high}} - f_{\text{low}}} \tag{1}$$

(where M is set to be greater than $T_{\text{ref}} f_{\text{low}}$), so each rising edge of the reference is always followed by a rising edge of

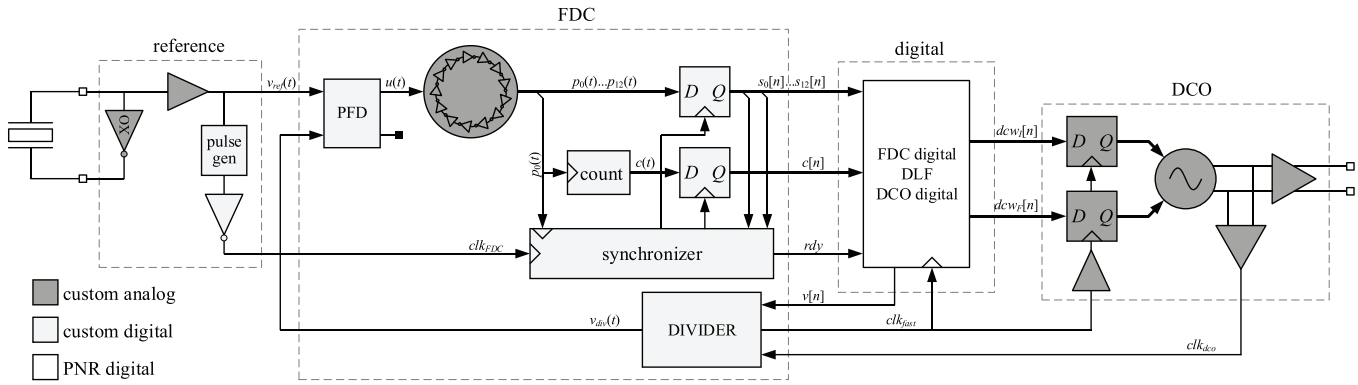

Fig. 6. Block diagram of the FDC-PLL showing the four supply domains with dashed boxes.

the divider output. This implies that the average frequency of the divider output is $f_{\rm ref}$, so the average value of $v[n]$, and, hence, $y[n]$, must be $-\alpha$ when the PLL output frequency is $(N + \alpha)f_{\rm ref}$. As also proven in [50], the average frequency of the DMRO locks to $Mf_{\rm ref}$ as a byproduct of the $\Delta\Sigma$ FDC's operation. Therefore, the DMRO is not a free-running oscillator; it is locked to a fixed integer multiple of the reference frequency, which minimizes the potential for fractional spurs.

The top structure in Fig. 5 is the well-known second-order $\Delta\Sigma$ modulator, and the bottom structure is an equivalent version of it. The $\Delta\Sigma$ FDC equivalence to a $\Delta\Sigma$ modulator can be seen from the bottom $\Delta\Sigma$ modulator in Fig. 5. The $2 - z^{-1}$ block of the $\Delta\Sigma$ modulator is implemented by the $2-z^{-1}$ block in the $\Delta\Sigma$ FDC. The second accumulator and quantizer are implemented by the integration and quantization performed by the DMRO and ring phase calculator as described above. The divider's output frequency varies in proportion to both $v[n]$ and the PLL's frequency error, i.e., its deviation from $(N + \alpha)f_{\rm ref}$, so the divider's output phase and, consequently, the width of the $u(t)$ pulse during the nth $\rm clk_{FDC}$ period is proportional to the integral of both $v[n]$ and the PLL's frequency error. This integration operation implements the first accumulator in the $\Delta\Sigma$ modulator.

Relationship to TDCs With Quantization Noise Shaping and First-Order FDCs: A significant aspect of the above-mentioned $\Delta\Sigma$ modulator equivalence is that the $\Delta\Sigma$ FDC inherits the self-dithering property of a second-order $\Delta\Sigma$ modulator, which suppresses spurious tones that would otherwise occur in its quantization noise [47], [48], [51], [54]. In contrast, previously published gated ring oscillator (GRO) and switched ring oscillator (SRO) time-to-digital converters (TDCs) and previously-published first-order FDCs that have been used in PLLs are only equivalent to first-order $\Delta\Sigma$ modulators, which are notorious for having quantization noise with large spurious tones [28]–[33].

In the prototype IC, $f_{\rm low} \neq 0$ so the DMRO is equivalent to an SRO (but it is not part of an SRO-TDC). However, the FDC-PLL architecture also works with $f_{\rm low} = 0$, in which case the DMRO would be equivalent to a GRO. The term DMRO was used for the ring oscillator in this paper for consistency with [50] which presents general mathematical results applicable to either case. In the following, SROs and GROs are collectively referred to as DMROs.

Previously published digital PLLs based on SRO-TDCs or GRO-TDCs, i.e., DMRO-TDCs, each have a divider controlled by a digital $\Delta\Sigma$ modulator such that its average modulus is $N + \alpha$ as in an analog PLL, and the sampled phase of the DMRO is subjected to a $1 - z^{-1}$ block to estimate the DMRO's instantaneous frequency and first-order shape its quantization noise. Therefore, each such PLL has quantization noise from both the digital $\Delta\Sigma$ modulator and the DMRO, whereas the FDC-PLL only has quantization noise from the DMRO. Furthermore, in a DMRO-TDC based PLL the DMRO is free-running whereas in the FDC-PLL the DMRO is locked to a multiple of the reference frequency as described in the previous subsection.

III. Implementation Details

Fig. 6 shows the FDC-PLL without its test logic or programming interface. The only blocks which are fully custom analog are the crystal oscillator (XO), DMRO, and DCO. The custom digital blocks are comprised entirely of standard cells, but manually laid out; manual versus automated layout gives better control over parasitics for improved speed and matching. The ring phase calculator block, the $2 - z^{-1}$ block, the DLF, and a portion of the DCO's control logic are all contained in the place and route (PNR) digital.

A. Clocking and Timing

$\Delta\Sigma$ FDC Timing: The $\Delta\Sigma$ FDC requires the divider to operate such that within one period of $\rm clk_{FDC}$ the $u(t)$ pulse rises and falls, the DMRO is sampled, and $y[n]$ and $v[n]$ are computed, all in time for the divider to be loaded for the next reference period. The DMRO's phase sample-rate is the reference frequency, and while it is functionally possible to sample the DMRO on the rising reference edge while $u(t)$ is high and the DMRO frequency is $f_{\rm high}$, it is much simpler timing-wise to sample it on the falling reference edge when $u(t)$ is low and the DMRO's frequency is $f_{\rm low}$. As shown in Fig. 7, $\rm clk_{FDC}$ is generated from delayed inverted reference edges; the purpose of the delay and high duty cycle is to save power in the synchronizer which is discussed in Section III-B. The $\Delta\Sigma$ FDC is not sensitive to jitter on this sampling edge because this noise is added after the second accumulator in the equivalent $\Delta\Sigma$ modulator, so it is subjected to the same highpass noise shaping as the quantization noise.

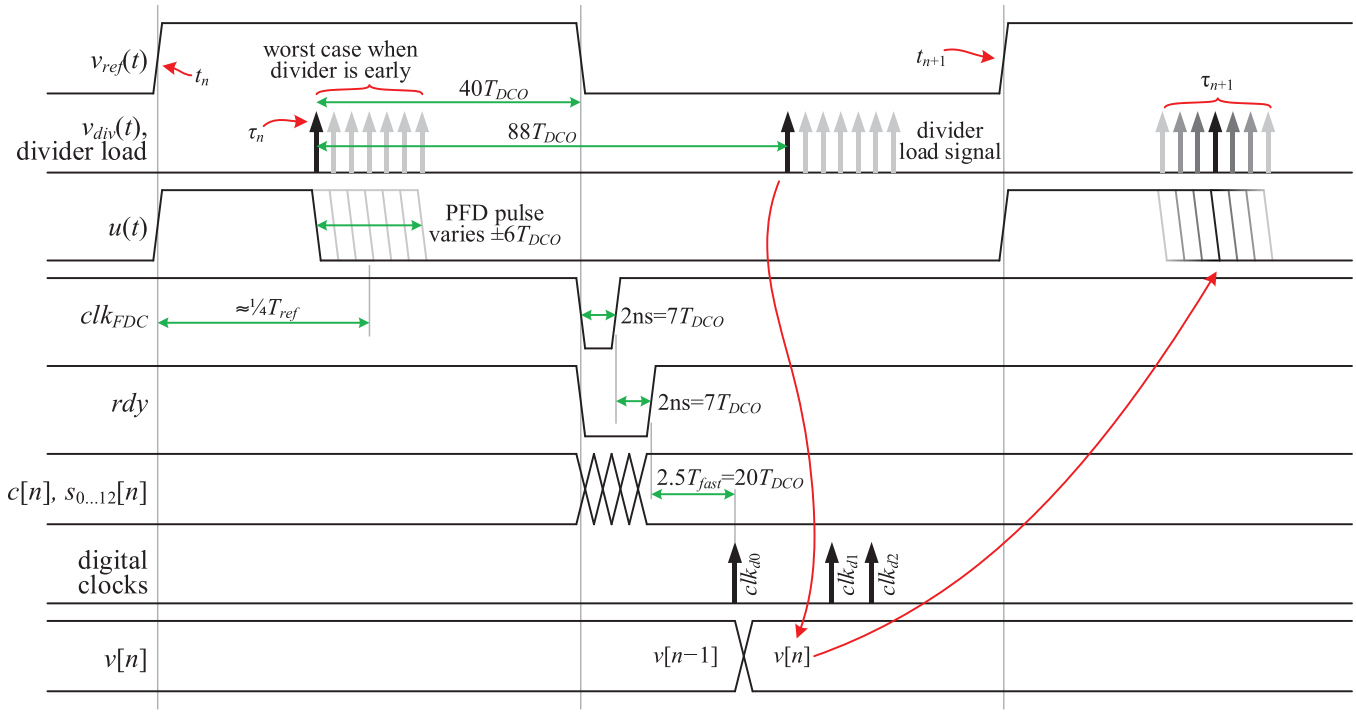

Fig. 7. FDC-PLL timing diagram.

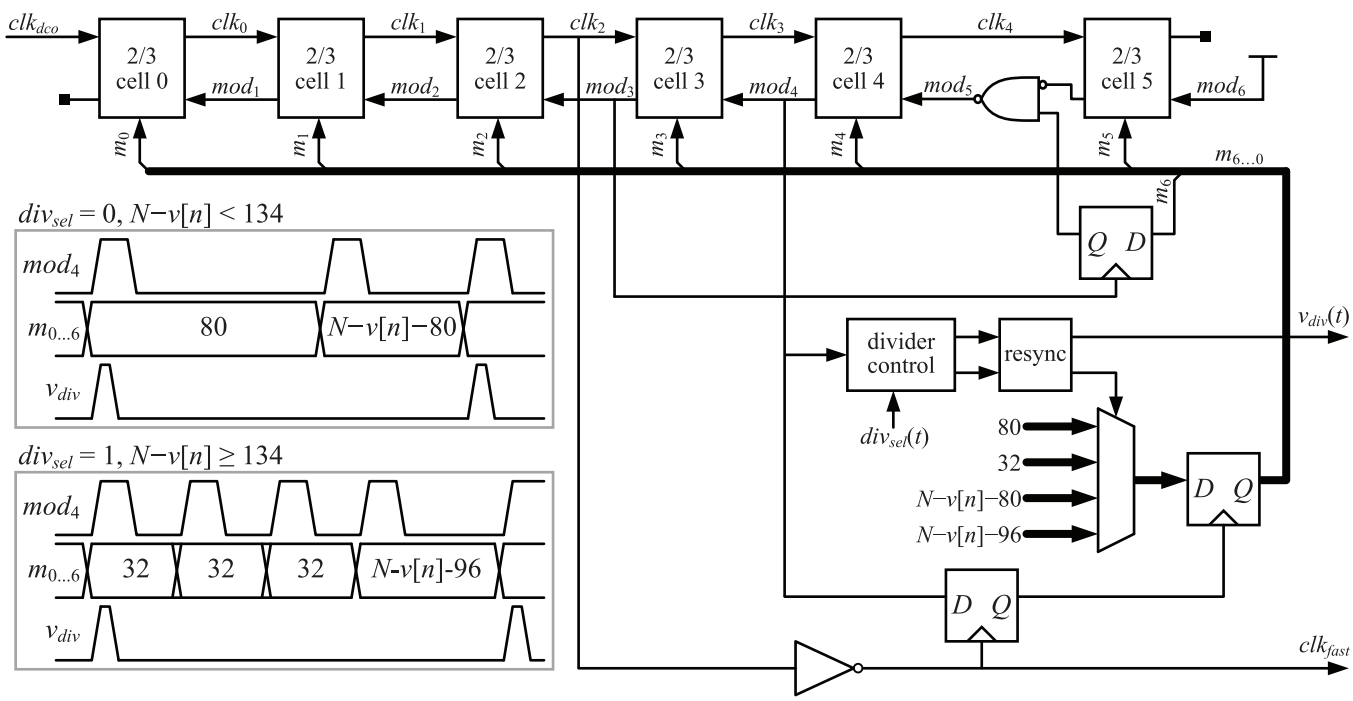

Fig. 8. Multi-modulus divider.

The timing bottleneck is generating $v[n]$ in time to affect the divider's next output edge. The value M is chosen according to (1) such that the average width, $T_{\bar{u}}$, of $u(t)$ is $1/4T_{\rm ref}$, the reason for which is explained in Section III-B. When $f_{\rm ref} = 26$ MHz and $f_{\rm pll} = 3.5$ GHz, $1/4T_{\rm ref} = 34T_{\rm DCO}$. It follows from the $\Delta\Sigma$ modulator equivalence that $|{\rm y}[n]| \leq 2$, and as $v[n]$ is the result of passing $y[n]$ through the $2 - z^{-1}$ block, this ensures that $|v[n]| \leq 6$. Thus, each $u(t)$ pulse is high for 28 to 40 DCO periods after the rising reference edge. If $u(t)$ is 28 DCO periods wide, then there are 40 DCO periods between $v_{\rm div}(t)$ and the falling edge of $v_{\rm ref}(t)$. Adding 2 ns for the $\rm clk_{FDC}$ delay, plus a worst case of 2 ns through the synchronizer means the data into the FDC digital is ready no later than 54 DCO periods into the current divider interval.

The divider shown in Fig. 8 is a multi-modulus divider similar to that presented in [55] but modified to allow the modulus to be loaded after the current division interval has begun. Rather than dividing by $N - v[n]$ with one chain of divide-by-2/3 cells as in

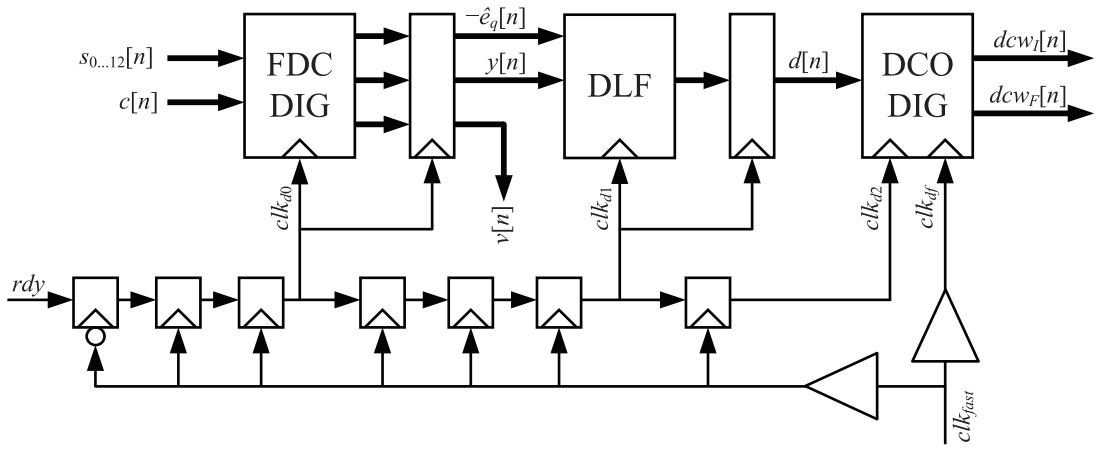

Fig. 9. High-level diagram of PNR digital sub-blocks and their clocking.

[55], the modulus is split into fixed and variable count intervals such that the modulus for the variable count interval need not be loaded until 8 DCO periods before the end of the prior fixed count interval. The divider has 6 divide-by-2/3 cells and control logic that re-uses the chain multiple times per division interval. When $\mathrm{div_{sel}} = 0$, the divider begins each period by dividing by 80. At the end of this interval the chain is reloaded with the modulus $N - v[n] - 80$, which completes the division. In this mode the divider can accommodate any modulus from 96 through 133. When $\mathrm{div_{sel}} = 1$, the divider divides by 32 three times for a total of 96, followed by the variable modulus division of $N - v[n] - 96$. In this mode the divider can accommodate any modulus from 134 to 159. In the FDC-PLL timing example in Fig. 7, $\mathrm{div_{sel}} = 1$ so the divider must be loaded before the $96\mathrm{th} - 8 = 88\mathrm{th}$ DCO period into its interval. The FDC digital (comprised of the ring phase calculator and $2 - z^{-1}$ block) has $88 - 54 = 34$ DCO periods to process the current DMRO sample and generate $v[n]$.

Digital Timing: A detailed diagram of the PNR digital is shown in Fig. 9. Its clock, $\mathrm{clk_{fast}}$, is generated by the divider's 3rd divide-by-2/3 cell output, which has an average frequency of $f_{\mathrm{dco}}/8$. The FDC digital, DLF, and a portion of the DCO digital (described in Section III-C) are all clocked at the reference rate by gated versions of $\mathrm{clk_{fast}}$. The synchronizer (described in Section III-B) generates the signal rdy which indicates that the sampled DMRO outputs and the ring phase calculator's sampled counter output are valid. The rdy signal is retimed in the PNR digital because it is asynchronous with respect to $\mathrm{clk_{fast}}$. This introduces a delay of 0.5–1.5 $\mathrm{clk_{fast}}$ periods. The output of the FDC digital is ready one $\mathrm{clk_{fast}}$ period after that, for a total of 1.5–2.5 $\mathrm{clk_{fast}}$ or 12–20 $\mathrm{clk_{DCO}}$ periods between synchronizer data to $v[n]$ output. In total, the $v[n]$ output is ready by the 65th–74th DCO period, well before the 88th period when the divider is loaded.

As shown in Figs. 7 and 9, the rdy signal continues to propagate down the register chain, gating the clock to the DLF and slow (i.e., f_{ref}-rate) portion of the DCO digital. This sequence completes before the next reference edge so that when $u(t)$ goes high, the majority of the PNR digital has completed switching. This results in a quiet environment for the DMRO, whose supply sensitivity was measured to be ten times higher when its frequency is f_{high} than when its frequency is f_{low}.

The DLF's output, $d[n]$, is synchronous with $\mathrm{clk_{fast}}$, so upsampling it within the DCO digital does not require resynchronization. The DCO digital outputs are retimed by a set of flip-flops near the DCO's FCE elements that are powered by the DCO's supply. These flip-flops are clocked by a version of $\mathrm{clk_{fast}}$ that is passed directly from the divider to the DCO to minimize jitter.

B. FDC Design

PFD: The $\Delta\Sigma$ FDC's PFD is identical to the tristate PFD commonly used in analog PLLs [56] except that it has been modified so that its output can only be high when $v_{\mathrm{ref}}(t)$ is high. This modification forces $u(t) = 0$ in the second half of each reference period, ensuring that the DMRO frequency is f_{low} when the DMRO outputs and ring phase calculator's counter are sampled. As mentioned above, the average width of $u(t)$ is around $1/4T_{\mathrm{ref}}$, so this modification has no effect on normal operation.

DMRO: The DMRO circuit is shown in Fig. 10. It uses the multi-path technique presented in [57], [58] to maximize the number of ring stages under the constraint that $f_{\mathrm{high}} - f_{\mathrm{low}} = f_{\mathrm{pll}}$, because the magnitude of $e_{Rq}[n]$ defined in Section II is inversely proportional to the number of stages. Each stage has three inputs: the output of the prior stage, the output from three stages back, and the output from five stages back. The transistors are sized so that the output from five stages back has the dominant influence on the current stage output. This shortens the effective path around the ring to be closer to a 3-stage oscillator. The other paths have a phase-interpolation effect to ensure that the propagation of edges around the ring is sequential. Without the multi-path technique, a conventional 13-stage ring oscillator in this IC technology would not achieve a high enough frequency to satisfy $f_{\mathrm{high}} - f_{\mathrm{low}} = f_{\mathrm{pll}}$.

DMRO tuning is achieved by current starving the core oscillator. The DMRO's low frequency, f_{low}, is tunable from 0.4–3.4 GHz by two SPI-controllable 4 bit resistor arrays, one between VDD and the core, and the other between the core and ground. Its high frequency, f_{high}, is controlled in the same way,

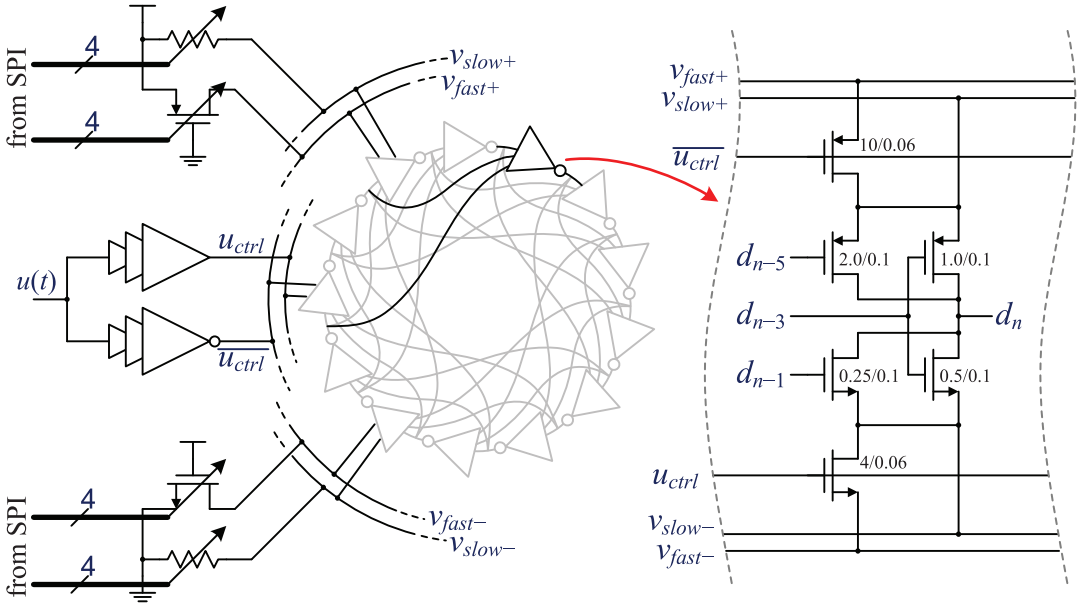

Fig. 10. DMRO circuit diagram with details of one of 13 identical stages.

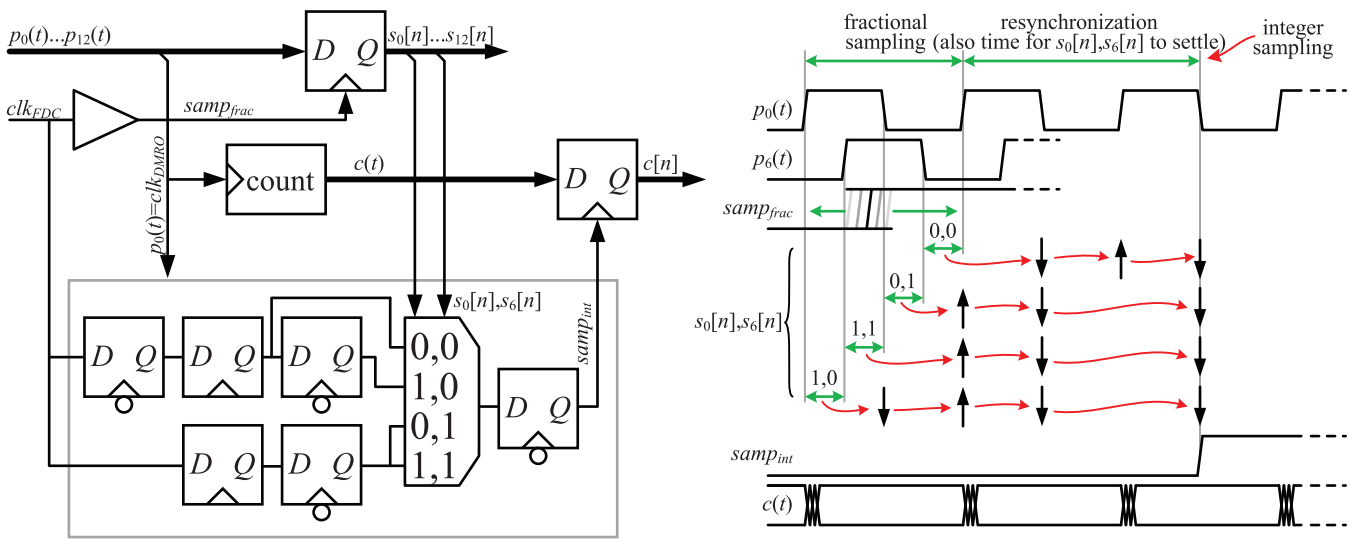

Fig. 11. DMRO sampling synchronizer and associated timing diagram.

except transistors in triode are used in place of resistors. Four bit tuning gives an f_{high} range of 1.8–5.1 GHz. The $u(t)$ signal is buffered and drives switches that connect the MOS array to the core, bypassing the resistor array, so as to modulate the DMRO between f_{low} and f_{high}.

A well-known property of charge-pump based analog PLLs is their low sensitivity to non-ideal charge pump switching transients provided that the charge pump current is allowed to fully settle between transient events, and that the rising and falling transient shapes are independent of when the current sources are turned on or off, respectively. For the same reasons, non-ideal DMRO frequency transients between f_{high} and f_{low} do not degrade the FDC-PLL's performance provided the DMRO frequency is allowed to fully settle before $u(t)$ transitions or the DMRO is sampled, and that the rising and falling frequency transient shapes are independent of the times of the rising and falling edges of $u(t)$, respectively. By setting M, f_{high} and f_{low} so that $u(t)$ is on average $1/4T_{\text{ref}}$ wide via (1), the settling time available for both the rising and falling frequency transients is maximized. Simulations show that the DMRO deviates from its ideal linear behavior by $\pm 0.35\%$, which results in in-band fractional spurs no great than -70 dBc.

Ring Phase Calculator Phase Sampling and Synchronizer: Fig. 11 shows the circuit and timing diagrams of the synchronizer mentioned in Section II, which is a variant of that presented in [36]. The synchronizer solves two problems which arise from asynchronous sampling of the DMRO by clk_{FDC}. The first is that inevitable timing skew between the counter sampler and DMRO sampler will cause glitches near counter increments because the counter will not increment at the exact moment the DMRO phase wraps. The second is that even if the counter and DMRO sampling paths are aligned, if the binary counter is sampled while it is incrementing, hugely incorrect sampled values may result.

The $\mathrm{samp}_{\mathrm{frac}}$ signal is a buffered version of $\mathrm{clk}_{\mathrm{FDC}}$ that samples the DMRO's phases $p_0(t), p_1(t), \ldots, p_{12}(t)$ to produce $s_0[n], \ldots, s_{12}[n]$. Unlike the binary counter, only one of the $p_0(t), \ldots, p_{12}(t)$ outputs transitions at a time, and incorrect samples of the actively transitioning output result in a decoded phase that is at most one 1/26th of a period (one fractional quantization step) in error. Since the metastable region of the sampling flip-flops is much narrower than a DMRO's stage delay when oscillating at frequency f_{low}, incorrect sampling is only likely to occur when the DMRO's phase is near a boundary between quantization levels, so the actual error due to an incorrect sample is much smaller than a fractional quantization step.

The pair of DMRO phases $p_0(t)$ and $p_6(t)$ are roughly in quadrature, so the pair of samples $(s_0[n], s_6[n])$ determine in which of roughly four equal parts of a $\mathrm{clk}_{\mathrm{DMRO}}$ period the $\mathrm{samp}_{\mathrm{frac}}$ rising edge occurs. Two delay lines clocked on $\mathrm{clk}_{\mathrm{DMRO}}$ sample $\mathrm{clk}_{\mathrm{FDC}}$, one starting with a rising edge and the other with the falling edge. Based on which of the four $\mathrm{clk}_{\mathrm{DMRO}}$ period sub-intervals the $\mathrm{samp}_{\mathrm{frac}}$ rising edge has arrived, the delay line which sampled $\mathrm{clk}_{\mathrm{FDC}}$ furthest from its rising edge is selected. The delay lines lengths are such that the generated $\mathrm{samp}_{\mathrm{int}}$ edge is always 1.5 $\mathrm{clk}_{\mathrm{DMRO}}$ periods after the $\mathrm{clk}_{\mathrm{DMRO}}$ period in which the $\mathrm{samp}_{\mathrm{frac}}$ rising edge arrived, allowing the samples $s_0[n]$ and $s_6[n]$ to settle before the MUX decision is required; this adds a constant offset to $c[n]$, which is irrelevant because $c[n]$ is first differenced in the ring phase calculator.

By using $(s_0[n], s_6[n])$ to determine where to sample the counter, the synchronizer is not sensitive to timing skew between $\mathrm{clk}_{\mathrm{DMRO}}$ and $\mathrm{clk}_{\mathrm{FDC}}$ up to a quarter of a $\mathrm{clk}_{\mathrm{DMRO}}$ period, T_{DMRO}. For example, if $\mathrm{samp}_{\mathrm{frac}}$ is delayed relative to $\mathrm{clk}_{\mathrm{FDC}}$, then if $\mathrm{clk}_{\mathrm{FDC}}$ lands in the later part of the (0, 1) interval, the samples $(s_0[n], s_6[n])$ may be (0, 0) rather than (1, 0). In this case sampling first with the falling edge rather than the rising edge still gives the correct result, because if the timing skew is less than one quarter of a DMRO period there are no falling $\mathrm{clk}_{\mathrm{DMRO}}$ edges between $\mathrm{clk}_{\mathrm{FDC}}$ and $\mathrm{samp}_{\mathrm{frac}}$. By inserting replica delays and careful layout, $1/4T_{\mathrm{DMRO}}$ delay matching is easy to achieve.

Retiming $\mathrm{samp}_{\mathrm{int}}$ to the falling edge of $\mathrm{clk}_{\mathrm{DMRO}}$ is the first step toward solving the second problem of sampling the binary counter, since the sampling is now synchronous. However this only allows the counter $1/2T_{\mathrm{DMRO}}$ to propagate each count. Fig. 12 shows how the binary ripple counter is "wave" sampled sequentially wherein the propagation speed of the wave is designed to approximately equal the propagation speed of the ripple down the counter bits. In a ripple counter, the Q output of one flip-flop is connected to the clk input of the following flip-flop. The propagation speed of an MSB transition, e.g., 0111... to 1000..., is therefore the cascade of the clk to Q delays of the flip-flops. By building a delay chain out of simplified flip-flops which mimic the clk to Q delay of a full flip flop, the sampling edge can propagate at approximately the same speed as the data through the ripple counter, which reduces the effective propagation delay of the counter to the difference between the total ripple delay and the total wave delay.

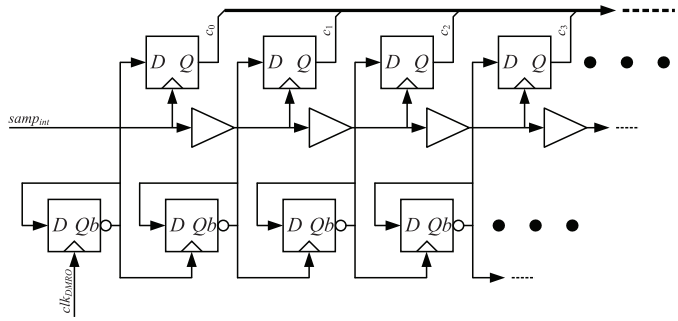

Fig. 12. Binary ripple counter and wave sampling scheme.

C. DCO

The DCO and its control circuitry are shown in Fig. 13. The DLF controls its frequency through two banks of frequency control elements (FCEs) of the type presented in [48]. Each unit-weight FCE creates a capacitance step of 32 aF which is equivalent to a 27 kHz frequency step at 3.5 GHz. The details of the integer boundary avoider are described shortly, but its $c_I[n]$ and $x_F[n]$ outputs represent information from the integer and fractional parts, respectively, of the DLF's 14 bit output, $d[n]$. The integer part is encoded to drive the *slow FCE bank*, a 2-2-2 ... 2-1 weighted array of 63 FCEs updated at the reference rate. The fractional part is upsampled at an eighth of the DCO frequency and requantized by a second-order digital $\Delta\Sigma$ modulator into a 5-level sequence similarly to [6]. This sequence is scrambled by a DEM encoder to produce $\mathrm{dcw}_F[n]$ as in [48], which drives the *fast FCE bank* comprised of four unit-weight FCEs. The LSB of $c_I[\mathrm{n}]$ has a frequency weight of one FCE (27 kHz), and the integer and fractional parts of $d[n]$ have bit-widths of 6 and 8, respectively, so the PLL controls the DCO over a range of $63 \cdot 27$ kHz $= 1.7$ MHz with a minimum step size of $2^{-8} \cdot 27$ kHz $= 105$ Hz.

A binary-weighted capacitor array controlled via the SPI interface is in parallel with the slow and fast FCE banks. It has 12 bits of tuning over a frequency range of 2.8–3.5 GHz, with a 400 kHz step size at 3.5 GHz. The main inductor is a custom 2-turn 1 nH center-tapped coil, and the regeneration is provided by a cross-coupled thick-oxide NMOS pair with a tail resonant tank as in [59] and triode-MOS tail source. The inductors, capacitors, and all metal routing were designed and extracted using the EMX 3D field solver.

Integer Boundary Avoider: When the PLL is locked, any slow variation in the DCO's frequency due to $1/f$ noise and supply and temperature changes will be tracked by the loop and compensated by a restoring change in $d[n]$. Inevitably $d[n]$ will wander toward an integer boundary, i.e., the point at which $x_F[n]$ wraps from 0.111... to 0 or vice versa and $c_I[n]$ increments or decrements correspondingly. When this happens, the wrapping of $x_F[n]$ will impart a frequency change equal to the average frequency step of the four FCEs in the fast bank. Simultaneously, the increment or decrement of the slow bank will impart a frequency change equal to the frequency step of the particular slow element that is switched. Because the LSB resolution of $d[n]$ is 105 Hz, the FCEs in the slow bank must match the average of the FCEs in the fast bank to better than 105 Hz, which is 2^8 times smaller than the FCE's 27 kHz frequency

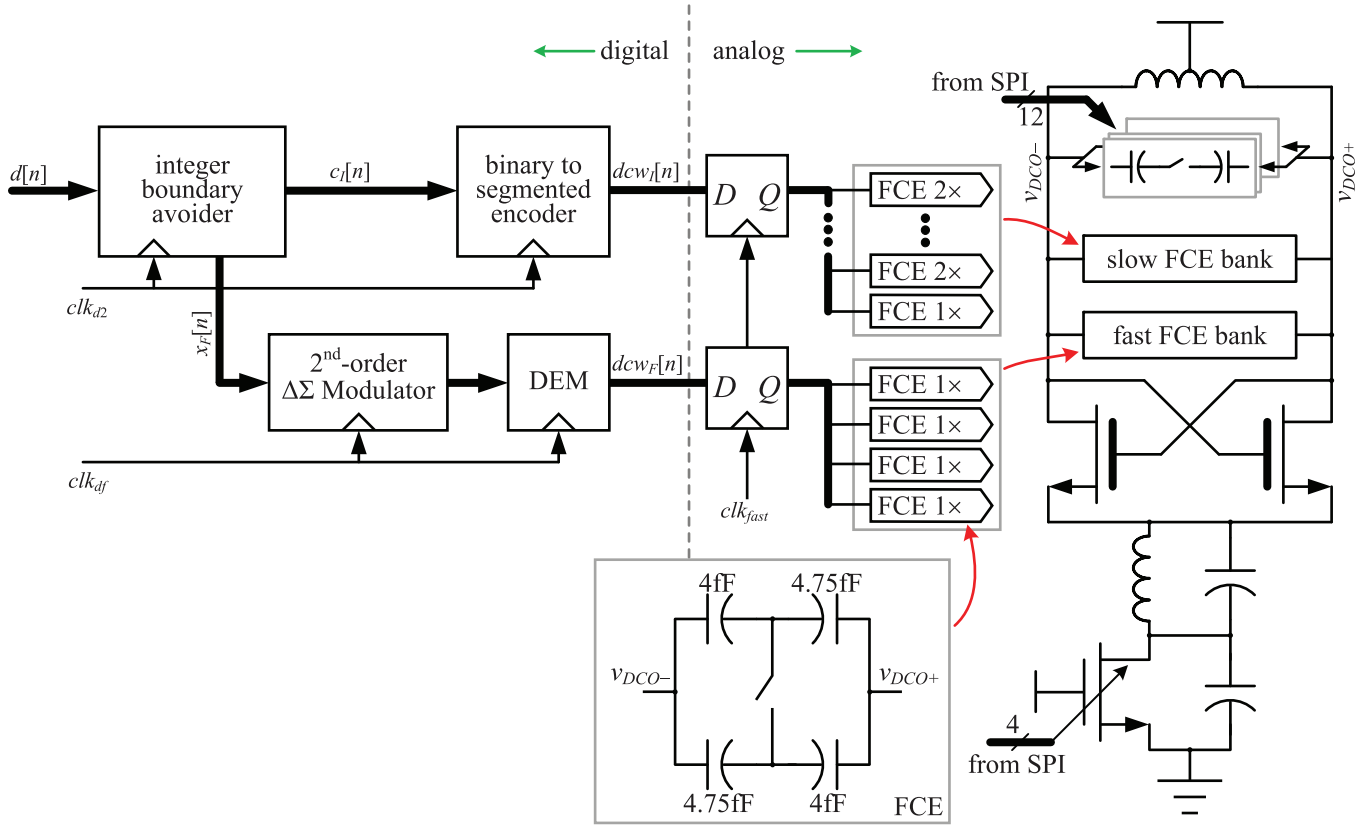

Fig. 13. DCO block diagram showing partition between custom analog and digital.

step. Achieving 8 bits of matching from 32 aF capacitors is impossible, so in practice large frequency glitches are injected each time an integer boundary is crossed. These glitches can create local non-monotonicites at integer boundaries that lead to limit-cycle-like oscillations whenever the boundary is crossed that degrade the PLL's phase noise. Fig. 14 shows a simulation of the FDC-PLL that demonstrates the effect this can have on phase noise for the FCE matching expected in the prototype IC.

The integer boundary avoider in Fig. 15 mitigates this problem by minimizing the change of $c_I[n]$ so as to reduce the number of physical integer boundary crossings. As shown, the DLF output is split into its integer and fractional parts, $d_I[n]$ and $d_F[n]$ such that $d[n] = d_I[n] + 2^{-8}d_F[n]$. When the AND gate output is high, it subtracts one from the integer path, and when it is low, it subtracts 2^8 from the fractional path; thus the sum $c_F[n] + 2^{-8}x_F[n]$ always equals $d[n]-1$. This doubles the range required of the $\Delta\Sigma$ modulator to $[-1, 1)$, which is why the fast FCE bank has four rather than three FCE elements. Swapping -1 from the integer to the fractional path and vice versa swaps which portion of the $\Delta\Sigma$ modulator's range—either $[-1, 0)$ or $[0, 1)$—is used. The logic sets the AND output such that the $\Delta\Sigma$ modulator's "0" point always straddles the previously crossed boundary; in this manner, re-crossings of a just-crossed integer boundary are handled by the $\Delta\Sigma$ modulator's range rather than the integer part.

The waveforms in Fig. 16 show the possible scenarios. Starting with the thick trace, when $d[n]$ crosses boundary k in the rising direction at time T_2, $c_I[n]$ changes and the slow bank increments; however, subsequent re-crossings of k at times T_3

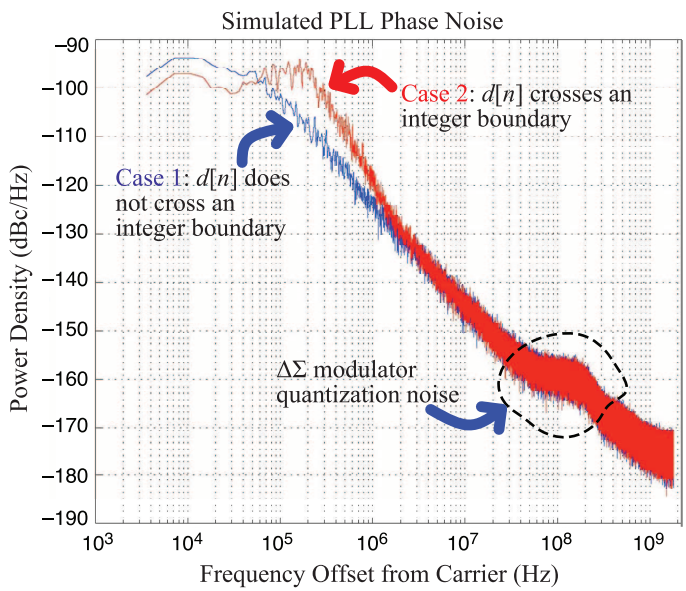

Fig. 14. Increased phase noise due to limit cycles near integer FCE boundaries in the DCO.

and T_4 do not affect $c_I[n]$, because only $x_F[n]$ changes. It is only if $d[n]$ were to cross $k + 1$ (the upper dashed trace) or $k-1$ (the lower dotted trace) would $c_I[n]$ change, and again in the upper dashed example, when the $k + 1$ is crossed for the second time at T_4, $x_F[n]$ is changed instead of $c_I[n]$.

The integer boundary avoider therefore prevents any hysteresis around transitions in the slow bank, and reduces the

TABLE I
AREA AND POWER BREAKDOWN OF THE IC

	Area (mm^2)	Power (mW)
PNR digital	0.07	4.6
XO and reference buffers	0.005	0.3
DCO and output buffer	0.21	9.0
ΔΣ FDC	0.02	1.7
Decoupling capacitance	0.035	-
Total	**0.34**	**15.6**

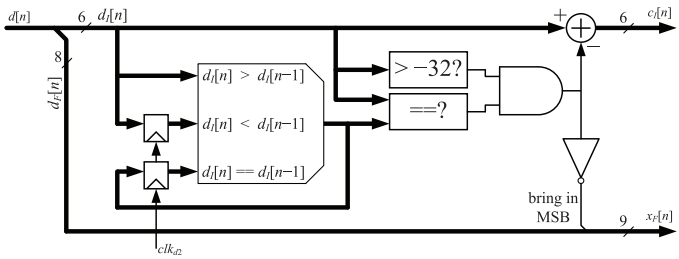

Fig. 15. Integer boundary avoider circuit.

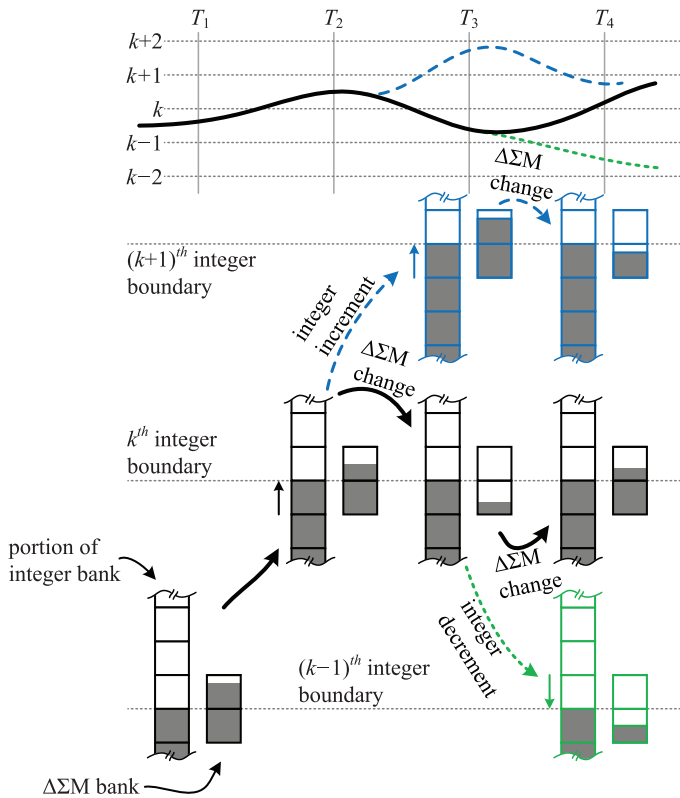

Fig. 16. Example of integer boundary avoider operation.

problem to occasional single-event disturbances in $d[n]$, on the order of the FCE mismatches. An alternative solution to this problem presented in [60] relies on foreground measurement and correction of the FCE mismatches using training sequences. While this technique does correct for the mismatches rather than just minimize their occurrences, it requires a calibration lasting on the order of minutes to fully characterize the whole FCE bank [61].

D. Power Distribution

The dotted boxes in Fig. 6 show the four separate power domains on the die: reference, FDC, digital, and DCO. The IC has one global ground provided by a low-impedance metal mesh that covers the active layout wherever possible. The use of a single ground simplified block-to-block communication as all signals are passed differentially with a ground shield, minimizing inter-supply current.

Each supply is heavily filtered with passive *RC* networks occupying any unused layout area, and the FDC supply was further sub-divided into four additional *RC*-filtered domains: PFD, divider, DMRO, and all the DMRO sampling and synchronization logic. For all the supplies with the exception of the reference, the *RC*-filter bandwidths were insufficiently low to have an appreciable impact on the fundamental harmonics of their supply currents. However, they were highly effective at minimizing supply bounces due to bondwire ringing from impulsive currents, which would have otherwise coupled back into its own circuitry or inductively to nearby bondwires.

IV. MEASUREMENT RESULTS

The prototype IC contains the FDC-PLL in Fig. 6 as well as test circuitry to measure internal voltages, clocks, and register values. The SPI interface logic and programming registers occupy 20% of the PNR digital area. The IC was fabricated in ST

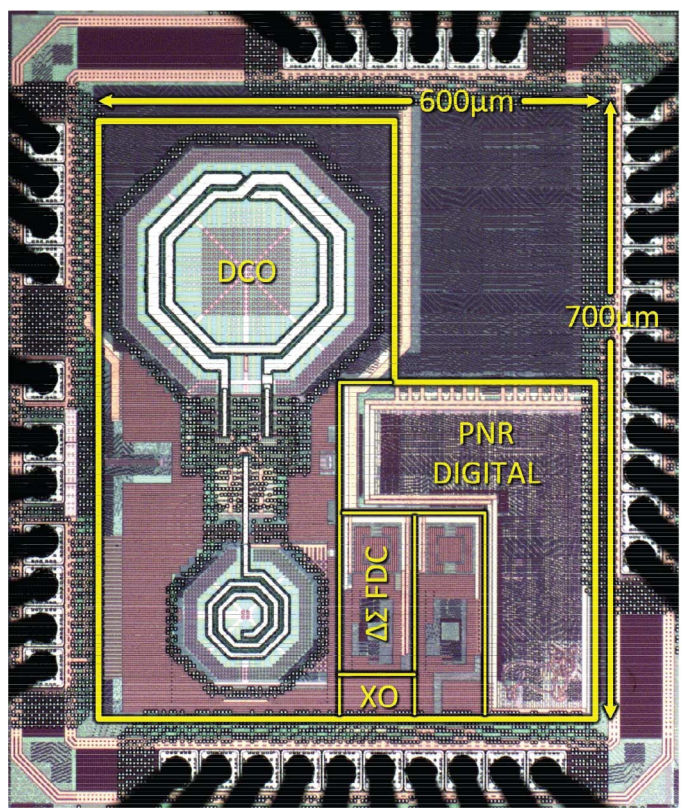

Fig. 17. Die photograph.

65 nm single-poly, seven-copper CMOS process, and makes use of the dual-oxide (LP and GP transistors both available) as well

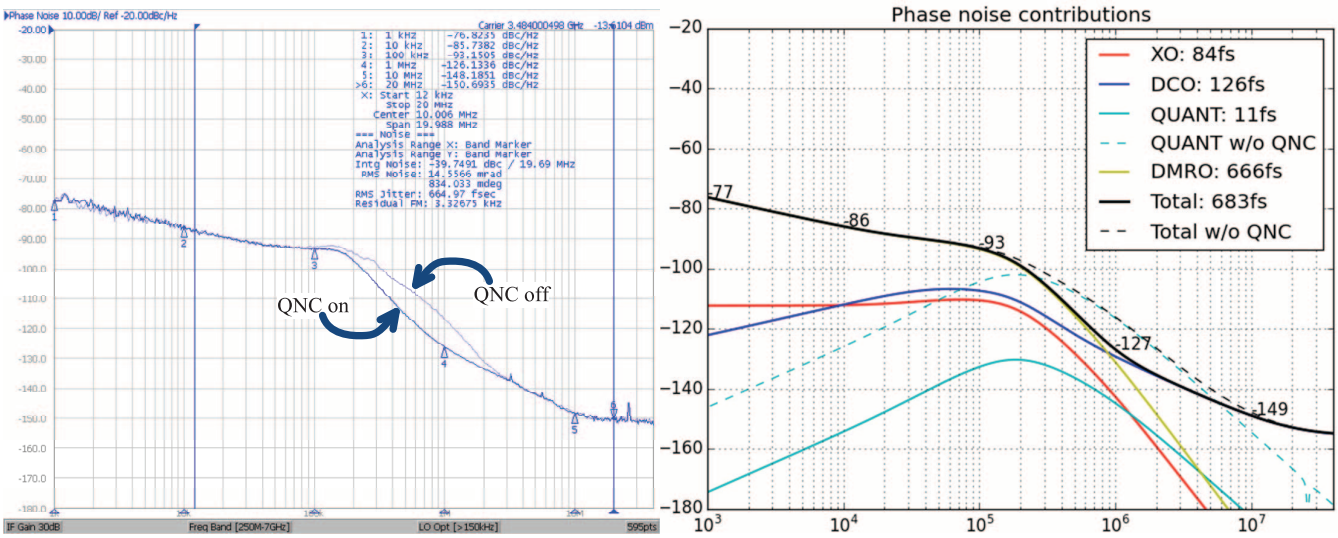

Fig. 18. Representative measured FDC-PLL phase noise for a 3.5 GHz output with a 400 Hz fractional frequency, and estimated phase noise contributions.

as high-resistivity poly process options. The die, which measures 1.0×1.3 mm^2, is shown in Fig. 17. The active area, which includes inductor density transition regions and all on-chip decoupling capacitance, is 0.34 mm^2. The area and power breakdowns are given in Table I.

The IC is packaged in a QFN32 package with a ground paddle. Sixteen copies of the IC were tested with a compression socket, of which four were damaged by a software bug that caused the IC to briefly receive 5 V during startup. Comprehensive measurements taken on the remaining 12 copies were all consistent. The presented spurious results were measured from one part because it was discovered that soldering the IC to the test board improved its spurious performance by 2–3 dB. This was determined by comparing the before-and-after-soldering measurement data for this particular part. The QFN footprint on the board was tinned rather than leveled and gold plated. It is suspected that unevenness in the tinning caused one or more pads to make poor contact when using the socket, which is corroborated by the fact that over-spec clamp-down pressure was required before the IC even drew current from the supply.

In addition to the IC, the test board contained an Abracon ABM8G 26 MHz crystal for the XO and a TDK HHM1583B1 wideband RF balun to match the differential output buffer to the measurement equipment. Power to the four supply domains was provided by Analog Devices ADP171 voltage regulators with parallel 10 μF X5R and 100 pF NP0 ceramic capacitors. While having independent supplies enabled characterization of individual blocks, for the measurements presented (with the exception of the DCO open-loop measurement discussed below) all the IC supply domains were connected together and driven with one regulator. The test board was connected to a motherboard that supplied power and USB communication to the measurement PC.

The phase noise measurements were taken with an Agilent E5052B signal source analyzer, and the spurious tone measurements were taken with an Agilent N9020A spectrum analyzer. In order to prevent unintentional alteration of data, all measurements, data collection, screen captures, and plot generation were performed using an automated suite of Python scripts.

TABLE II
DCO'S MEASURED PERFORMANCE AT EACH OUTPUT FREQUENCY EXTREME FROM A 1.0 V SUPPLY

Center frequency (GHz)		**2.8**	**3.5**
Supply current (mA)		9.4	8.3
Supply push (MHz/V)		-51	-68
1/*f* noise corner (kHz)		100	170
Phase noise (dBc/Hz)	**10 kHz**	-79.1	-76.3
	1 MHz	-129.5	-129.4
	10 MHz	-149.7	-150.1

The FDC-PLL's phase noise for a 3.5 GHz output with a 400 Hz fractional frequency offset is shown in Fig. 18. It is suspected that an output-power-limiting impedance mismatch limits the phase noise floor, as 20 MHz spot phase noise as low as -154 dBc/Hz has been observed in the lab with a high-quality SMA cable. However, for consistency with previously-taken measurements, the same generic SMA cable was used for all measurements. As shown, the PLL exceeds GSM phase noise requirements, which remain among the most difficult specifications to meet. By running multiple measurements with tweaked PLL configuration parameters, the phase noise contributions of all the individual blocks were extrapolated to produce the plot shown in Fig. 18. The DMRO's extrapolated phase noise has $1/f^3$ and $1/f^2$ contributions of -104 and -109 dBc/Hz at 1 MHz offset normalized to 1 GHz. The DMRO dominates the in-band noise, whose phase noise contribution to the output sees a band-pass transfer function [50]. While this level of noise is sufficient for this application, for other applications which may require wider loop bandwidths and/or lower in-band noise plateaus the DMRO phase noise would have to be improved. Given that the DMRO consumes less than 1.7 mW versus 9 mW for the DCO, DMRO power could be doubled for a 3 dB improvement in in-band noise with little impact on overall power consumption.

TABLE III
PERFORMANCE SUMMARY AND COMPARISON TABLE

	[7]	[11]	[20]	[21]	[24]	[25]	[28]	[31]	[35]	**This work**
Technology (nm)	90	130	65	65	65	65	130	28	65	65
Supply (V)	1.2	Not Stated	1.2	1.2	1.5/1.2	1.5	1.5	1.05	1.0	1.0
Power Consumption(1) (mW)	Not Stated	>40(4)	8.7	4.5	45	41.6	39	11.6	3.7	15.6
Area (mm^2)	Not Stated	0.86	0.44	0.22	0.6	0.7	0.95	0.22	0.22	0.35
Reference Frequency (MHz)	26	26	35	40	78	26	50	40	50	26
PLL Frequency (GHz)	3.6	3.6	3.5	3.6	4.0	1.8	3.7	2	4.5	3.5
Bandwidth (kHz)	40	50	3400	312	1000	800	500	250	750	140
In-band Phase Noise(2) (dBc/Hz)	-81 @20kHz	-79 @10kHz	-101 @1MHz	-101 @100kHz	-109 @200kHz	-102 @60kHz	-107 @400kHz	-97 @100kHz	-104 @100kHz	-93 @100kHz
Out-of-band Phase Noise(2) (1MHz, 3MHz, 20MHz) (dBc/Hz)	-117(3) - -152	-126 -136 -152	- - -129	- -121 -145	- - -149	- -122 -154	- -131 -149	-123 -137 -146	- -118 -126	-126 -138 -151
In-band Fractional Spur (dBc)	Not Stated	Not Stated	-58	-42	-40	-50	-42	Not Stated	-52	-60
Reference Spur (dBc)	-92	-84	-61	-72	-56	Not Stated	-65	-94	-69	-81

(1) Power and supply measurements are for core PLL circuitry, excluding RF output buffers
(2) Phase noise is normalized to a 3.5GHz PLL frequency
(3) Calculated by extending -20dB/decade phase noise slope from 400kHz measurement to 1MHz
(4) PLL draws 40mA from an unknown supply voltage in 130nm CMOS

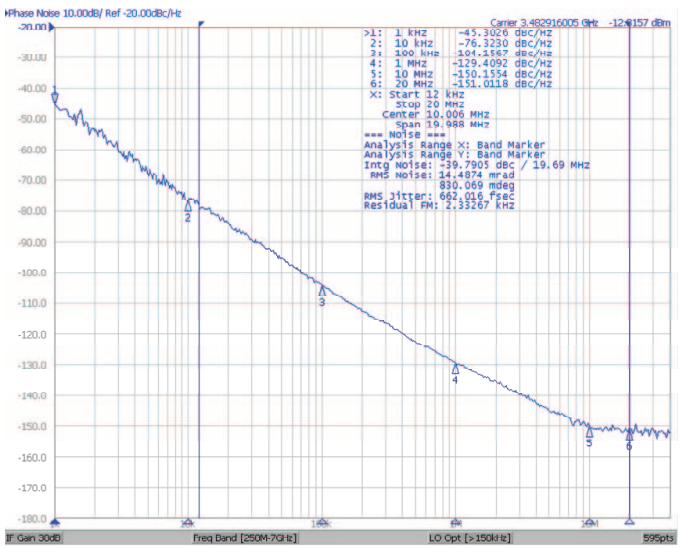

Fig. 19. Measured DCO phase noise at an output frequency of 3.5 GHz.

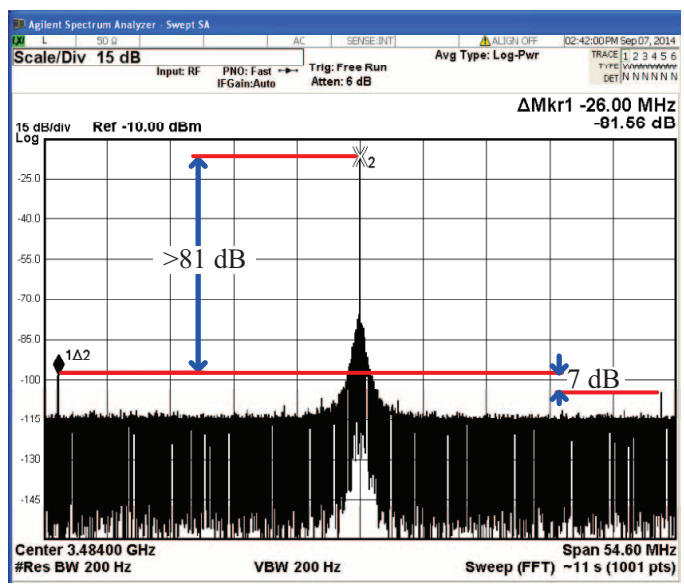

Fig. 20. Measured FDC-PLL output spectrum showing reference spur.

The open-loop DCO performance is listed in Table II, and its phase noise for a 3.5 GHz output is shown in Fig. 19. The DCO's true low-frequency noise was only visible after a 220 μF electrolytic capacitor was added in parallel with those already attached to the DCO supply regulator. For the closed-loop plot in Fig. 18, the PLL had sufficiently wide bandwidth to suppress the DCO regulator noise, making the electrolytic capacitor unnecessary. The oscillator core's low intrinsic supply sensitivity is a result of the low nonlinear parasitic capacitance on the oscillator nodes, due to the FCE's use of MOM rather than MOS unit capacitors [48].

The reference spur shown in Fig. 20 is −81 dBc. Due to the asymmetry of the negative and positive offset spurs, it is suspected that the origin of the −81 dBc spur is direct coupling, e.g., through bondwires, not upconversion within the PLL. Repeated sweeps of the spectrum analyzer showed the positive

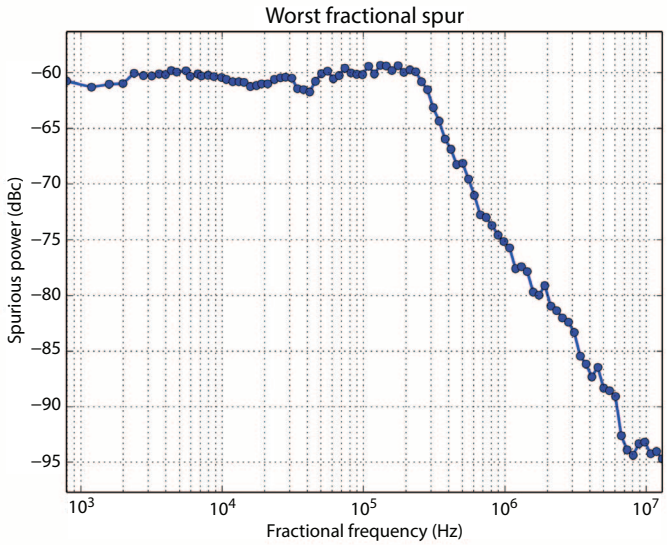

Fig. 21. The largest measured fractional spurious tones as a function of the PLL's fractional frequency setting.

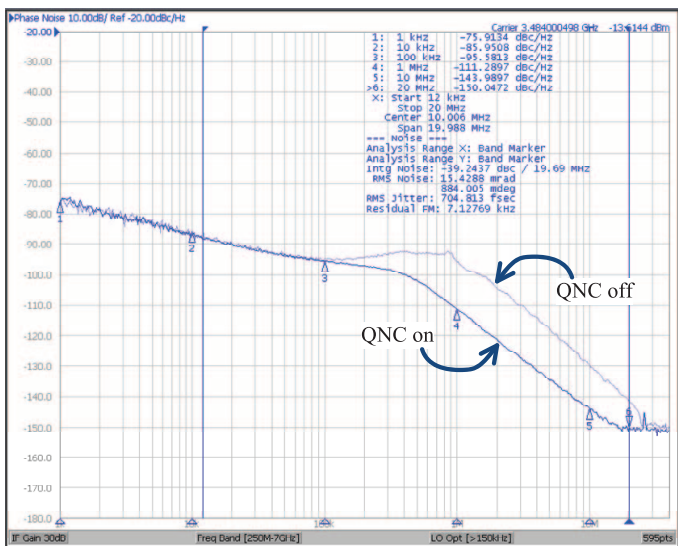

Fig. 22. FDC-PLL phase noise when configured with a 260 kHz bandwidth emphasizing the effect of QNC.

offset spur sometimes disappearing below the noise floor, while the −81 dBc negative-side reference spur remained constant. It is therefore a worst-case bound on reference spur performance. Spectrum analyzer averaging was disabled for this and all spur measurements.

The PLL's fractional frequency offset, α, was swept from 0 to 1/2 and the PLL's worst fractional spur is plotted for each value of α in Fig. 21. For this measurement, the spectrum analyzer's span, sweep time and resolution bandwidth were automatically adjusted for each value of α to ensure the noise floor was low enough to see spurs, and that five negative and positive harmonics of αf_{ref} were always visible. The worst fractional spur was always either the first or second, and neither exceeded −60 dBc.

To demonstrate the efficacy of QNC, Fig. 22 shows the PLL's phase noise with the bandwidth set to 260 kHz in order to allow more quantization noise into the PLL's output, and make its reduction by QNC more prominent. The phase noise suppression is less than the expected $20\log_{10}(26\ \text{phases}) = 28$ dB because when QNC is enabled, 1–10 MHz phase noise is dominated by DMRO and DCO noise rather than quantization noise. This plot verifies the theoretical operation of the ΔΣ FDC; the accumulated ΔΣ FDC's quantization noise is cancelled by first-differenced DMRO quantization noise, which is only possible if the ΔΣ FDC's quantization noise has second-order shaping. The plot also implicitly shows the level of matching between the DMRO phase taps, because any mismatch in DMRO elements would not be canceled by QNC.

The FDC-PLL's measured performance is summarized in Table III along with that of the best comparable PLLs published to date. As indicated, the FDC-PLL has excellent phase noise performance and low power consumption with the lowest supply voltage and best spurious tone performance in its class.[2] The PLL presented in [20] achieves spurious tone performance that is only 2 dB worse than that of the FDC-PLL, but it uses dither to suppress spurious tones at the expense of increased phase noise. Compared to other digital PLLs with similarly low phase noise, the FDC-PLL achieves an order of magnitude lower spurious tone power than previously reported.

Acknowledgment

The authors are grateful to Kevin Rivas for test motherboard development, Kelsey Curtis for layout artwork, Prof. Gabriel Rebeiz for access to measurement equipment, Integrand software for access to the EMX program, STMicroelectronics for IC fabrication and fruitful discussions, and Chris Mangelsdorf for helpful advice.

References

[1] B. Razavi, *Phase-Locking in High-Performance Systems: From Devices to Architectures*. New York, NY, USA: Wiley-Interscience, 2003.

[2] B. Miller and B. Conley, "A multiple modulator fractional divider," in *Proc. Annu. IEEE Symp. Frequency Control*, 1990, vol. 44, pp. 559–568.

[3] B. Miller and B. Conley, "A multiple modulator fractional divider," *IEEE Trans. Instrum. Meas.*, vol. 40, no. 3, pp. 578–583, Jun. 1991.

[4] T. A. Riley, M. A. Copeland, and T. A. Kwasniewski, "Delta-sigma modulation in fractional-*N* frequency synthesis," *IEEE J. Solid-State Circuits*, vol. 28, no. 5, pp. 553–559, May 1993.

[5] R. Best, *Phase-Locked Loops: Design, Simulation, and Applications*, 16th ed. New York, NY, USA: McGraw-Hill, 2007.

[6] R. B. 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, no. 12, pp. 2278–2291, Dec. 2004.

[7] R. B. Staszewski *et al.*, "All-digital PLL and transmitter for mobile phones," *IEEE J. Solid-State Circuits*, vol. 40, no. 12, pp. 2469–2482, Dec. 2005.

[8] K. Muhammad *et al.*, "The first fully integrated quad-band GSM/GPRS receiver in a 90-nm digital CMOS process," *IEEE J. Solid-State Circuits*, vol. 41, no. 8, pp. 1772–1783, Aug. 2006.

[9] R. Tonietto, E. Zuffetti, R. Castello, and I. Bietti, "A 3 MHz bandwidth low noise RF all digital PLL with 12 ps resolution time-to-digital converter," in *Proc. Eur. Solid-State Circuits Conf. (ESSCIRC)*, 2006, pp. 150–153.

[2]An often-quoted figure of merit (FoM) for PLLs is the Gao FoM defined in [62]. The Gao FoM for the FDC-PLL is 232 dB, which is lower than those of several recently published digitally PLLs [21], [31], [35], because the FDC-PLL was optimized for the GSM phase noise mask rather than low integrated jitter. If the goal had instead been to optimize the FDC-PLL's Gao FoM, the DCO power could have been reduced considerably (the DCO is a negligible jitter contributor as shown in Table II and Fig. 18).

[10] C. Weltin-Wu, E. Temporiti, D. Baldi, and F. Svelto, "A 3 GHz fractional-N all-digital PLL with precise time-to-digital converter calibration and mismatch correction," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2008, pp. 344–345.

[11] H. H. Chang, P.-Y. Wang, J.-H.C. Zhan, and H. Bing-Yu, "A fractional spur-free ADPLL with loop-gain calibration and phase-noise cancellation for GSM/GPRS/EDGE," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2008, pp. 200–201, 606.

[12] M. Lee, M. E. Heidari, and A. A. Abidi, "A low-noise wideband digital phase-locked loop based on a coarse-fine time-to-digital converter with subpicosecond resolution," *IEEE J. Solid-State Circuits*, vol. 44, no. 10, pp. 2808–2816, Oct. 2009.

[13] V. Kratyuk, P. K. Hanumolu, K. Ok, U. Moon, and K. Mayaram, "A digital PLL with a stochastic time-to-digital converter," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 56, no. 8, pp. 1612–1621, Aug. 2009.

[14] M. S.-W. Chen, D. Su, and S. Mehta, "A calibration-free 800 MHz fractional-N digital PLL with embedded TDC," *IEEE J. Solid-State Circuits*, vol. 45, no. 12, pp. 2819–2827, Dec. 2010.

[15] M. Zanuso, S. Levantino, C. Samori, and A. Lacaita, "A 3 MHz-BW 3.6 GHz digital fractional-N PLL with sub-gate-delay TDC, phase-interpolation divider, and digital mismatch cancellation," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2010, pp. 476–477.

[16] J. Borremans, K. Vengattaramane, V. Giannini, B. Debaillie, W. V. Thillo, and J. Craninckx, "A 86 MHz-to-12 GHz digital-intensive phase-modulated PLL for software-defined radios, using a 6 fJ/step TDC in 40 nm digital CMOS," *IEEE J. Solid-State Circuits*, vol. 45, no. 10, pp. 2116–2129, Oct. 2010.

[17] S. K. Lee, Y. H. Seo, Y. Suh, H. J. Park, and J. Y. Sim, "A 1 GHz ADPLL with a 1.25 ps minimum-resolution sub-exponent TDC in 0.18 m CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2010, pp. 482–483.

[18] T. Tokairin, M. Okada, M. Kitsunezuka, M. Tadashi, and M. Fukaishi, "A 2.1-to-2.8-GHz low-phase-noise all-digital frequency synthesizer with a time-windowed time-to-digital converter," *IEEE J. Solid-State Circuits*, vol. 45, no. 12, pp. 2582–2590, Dec. 2010.

[19] K. Waheed, K. M. Sheba, R. B. Staszewski, F. Dulger, and S. D. Vamvakos, "Spurious free time-to-digital conversion in an ADPLL using short dithering sequences," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 58, no. 9, pp. 2051–2060, Sep. 2011.

[20] E. Temporiti, C. Weltin-Wu, D. Baldi, M. Cusmai, and F. Svelto, "A 3.5 GHz wideband ADPLL with fractional spur suppression through TDC dithering and feedforward compensation," *IEEE J. Solid-State Circuits*, vol. 45, no. 12, pp. 2723–2736, Dec. 2010.

[21] D. Tasca, M. Zanuso, G. Marzin, S. Levantino, C. Samori, and A. L. Lacaita, "A 2.9-to-4.0 GHz fractional-N digital PLL with bang-bang phase detector and 560 fs rms integrated jitter at 4.5 mW power," *IEEE J. Solid-State Circuits*, vol. 46, no. 12, pp. 2745–2758, Dec. 2011.

[22] G. Marzin, S. Levantino, C. Samori, and A. Lacaita, "A 20 Mb/s phase modulator based on a 3.6 GHz digital PLL with -36 dB EVM at 5 mW power," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2012, pp. 342–343.

[23] J. Chen, L. Rong, F. Jonsson, G. Yang, and L.-R. Zheng, "The design of all-digital polar transmitter based on ADPLL and phase synchronized $\Delta\Sigma$ modulator," *IEEE J. Solid-State Circuits*, vol. 47, no. 5, pp. 1154–1164, May 2012.

[24] L. Vercesi, L. Fanori, F. De Bernardinis, A. Liscidini, and R. Castello, "A dither-less all digital PLL for cellular transmitters," *IEEE J. Solid-State Circuits*, vol. 47, no. 8, pp. 1908–1920, Aug. 2012.

[25] K. Takinami, R. Strandberg, P. C. P. Liang, G. L. G. de Mercey, T. Wong, and M. Hassibi, "A rotary-traveling-wave-oscillator-based all-digital PLL with a 32-phase embedded phase-to-digital converter in 65 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2011, pp. 100–102.

[26] K. Euda, T. Uozumi, R. Endo, T. Nakamura, T. Heima, and H. Sato, "A digital PLL with two-step closed-locking for multi-mode/multi-band SAW-less transmitter," in *Proc. IEEE Custom Integrated Circuits Conf. (CICC)*, 2012, pp. 1–4.

[27] B. Shen, G. Unruh, M. Lugthart, C. H. Lee, and M. Chambers, "An 8.5 mW, 0.07 mm^2 ADPLL in 28 nm CMOS with sub-ps resolution TDC and $<$230 fs RMS jitter," in *Symp. VLSI Circuits Dig.*, 2013, pp. C192–C193.

[28] C. Hsu, M. Z. Straayer, and M. H. Perrott, "A low-noise, wide-BW 3.6 GHz digital $\Delta\Sigma$ fractional-N frequency synthesizer with a noise-shaping time-to-digital converter and quantization noise cancellation," *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2776–2786, Dec. 2008.

[29] D.-W. Jee, Y.-H. Seo, H.-J. Park, and J.-Y. Sim, "A 2 GHz fractional-N digital PLL with 1 b noise shaping TDC," *IEEE J. Solid-State Circuits*, vol. 47, no. 4, pp. 875–883, Apr. 2012.

[30] C.-W. Yao and A. L. Willson, "A 2.8–3.2-GHz fractional-N digital PLL with ADC-assisted TDC and inductively coupled fine-tuning DCO," *IEEE J. Solid-State Circuits*, vol. 48, no. 3, pp. 698–710, Mar. 2013.

[31] M. A. Ferriss and M. P. Flynn, "A 14 mW fractional-N PLL modulator with a digital phase detector and frequency switching scheme," *IEEE J. Solid-State Circuits*, vol. 43, no. 11, pp. 2464–2471, Nov. 2008.

[32] L. Li, M. P. Flynn, and M. A. Ferriss, "A 5.8 GHz digital arbitrary phase-setting type II PLL in 65 nm CMOS with 2.25° resolution," in *Proc. IEEE Asian Solid-State Circuits Conf.*, 2012, pp. 317–320.

[33] M. Talegaonkar *et al.*, "A 4.4–5.4 GHz digital fractional-N PLL using $\Delta\Sigma$ frequency-to-digital converter," in *Symp. VLSI Circuits Dig.*, 2014.

[34] F.-W. Kuo *et al.*, "A 12 mW all-digital PLL based on class-F DCO for 4G phones in 28 nm CMOS," in *Symp. VLSI Circuits Dig.*, 2014.

[35] A. Elkholy, T. Anand, C. Woo-Seok, A. Elshazly, and P. Hanumolu, "A 3.7 mW low-noise wide-bandwidth 4.5 GHz digital fractional-N PLL using time amplifier-based TDC," *IEEE J. Solid-State Circuits*, vol. 50, no. 4, pp. 867–881, Apr. 2015.

[36] B. Staszewski, K. Waheed, F. Dulger, and O. Eliezer, "Spur-free multirate all-digital PLL for mobile phones in 65 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 46, no. 12, pp. 2904–2919, Dec. 2011.

[37] S. Levantino, G. Marzin, and C. Samori, "An adaptive pre-distortion technique to mitigate the DTC nonlinearity in digital PLLs," *IEEE J. Solid-State Circuits*, vol. 49, no. 8, pp. 1762–1772, Aug. 2014.

[38] E. Temporiti, G. Albasini, I. Bietti, R. Castello, and M. Colombo, "A 700 kHz bandwidth $\Sigma\Delta$ fractional synthesizer with spurs compensation and linearization techniques for WCDMA applications," *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1446–1454, Sep. 2004.

[39] S. E. Meninger and M. H. Perrott, "A 1 MHz bandwidth 3.6 GHz 0.18 μm CMOS fractional-N synthesizer utilizing a hybrid PFD/DAC structure for reduced broadband phase noise," *IEEE J. Solid-State Circuits*, vol. 41, no. 4, pp. 966–980, Apr. 2006.

[40] B. Zhang, P. E. Allen, and J. M. Huard, "A fast switching PLL frequency synthesizer with an on-chip passive discrete-time loop filter in 0.25-μ m CMOS," *IEEE J. Solid-State Circuits*, vol. 38, pp. 855–865, Jun. 2003.

[41] C. Park, O. Kim, and B. Kim, "A 1.8-GHz self-calibrated phase-locked loop with precise I/Q matching," *IEEE J. Solid-State Circuits*, vol. 36, no. 5, pp. 777–783, May 2001.

[42] Y.-C. Yang, S.-A. Yu, Y.-H. Liu, T. Wang, and S.-S. Lu, "A quantization noise suppression technique for $\Delta\Sigma$ fractional-N frequency synthesizers," *IEEE J. Solid-State Circuits*, vol. 41, no. 11, pp. 2500–2511, Nov. 2006.

[43] B. De Muer and M. S. J. Steyaert, "A CMOS monolithic $\Delta\Sigma$-controlled fractional-N frequency synthesizer for DCS-1800," *IEEE J. Solid-State Circuits*, vol. 37, no. 7, pp. 835–844, Jul. 2002.

[44] K. Wang, A. Swaminathan, and I. Galton, "Spurious-tone suppression techniques applied to a wide-bandwidth 2.4 GHz fractional-N PLL," *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2787–2797, Dec. 2008.

[45] H. Hedayati, W. Khalil, and B. Bakkaloglu, "A 1 MHz bandwidth, 6 GHz 0.18 μm CMOS type-I $\Delta\Sigma$ fractional-N synthesizer for WiMAX applications," *IEEE J. Solid-State Circuits*, vol. 44, no. 12, pp. 3244–3252, Dec. 2009.

[46] W. T. Bax and M. A. Copeland, "A GMSK modulator using a $\Delta\Sigma$ frequency discriminator-based synthesizer," *IEEE J. Solid-State Circuits*, vol. 36, no. 8, pp. 1218–1227, Aug. 2001.

[47] C. Venerus and I. Galton, "Delta-Sigma FDC based fractional-N PLLs," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 60, no. 5, pp. 1274–1285, May 2013.

[48] C. Venerus and I. Galton, "A TDC-free mostly-digital FDC-PLL frequency synthesizer with a 2.8–3.5 GHz DCO," *IEEE J. Solid-State Circuits*, vol. 50, no. 2, pp. 450–463, Feb. 2015.

[49] C. Weltin-Wu, G. Zhao, and I. Galton, "A highly-digital frequency synthesizer using ring oscillator frequency-to-digital conversion and noise cancellation," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2015.

[50] C. Weltin-Wu, E. Familier, and I. Galton, "A linearized model for the design of fractional-N PLLs based on dual-mode ring oscillator FDCs," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 62, no. 8, pp. 2013–2023, Aug. 2015.

[51] R. Schreier and G. C. Temes, *Understanding Delta-Sigma Data Converters*. New York, NY, USA: Wiley, 2005.

[52] S. Pamarti, L. Jansson, and I. Galton, "A wideband 2.4-GHz delta-sigma fractional-*N* PLL with 1-Mb/s in-loop modulation," *IEEE J. Solid-State Circuits*, vol. 39, no. 1, pp. 49–62, Jan. 2004.
[53] A. Swaminathan, K. Wang, and I. Galton, "A wide-bandwidth 2.4 GHz ISM-band fractional-*N* PLL with adaptive phase-noise cancellation," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2639–2650, Dec. 2007.
[54] I. Galton, "Granular quantization noise in a class of Delta-Sigma modulators," *IEEE Trans. Inf. Theory*, vol. 40, no. 3, pp. 848–859, May 1994.
[55] C. S. Vaucher, I. Ferencic, M. Locher, S. Sedvallson, U. Voegeli, and Z. Wang, "A family of low-power truly modular programmable dividers in standard 0.35-μm CMOS technology," *IEEE J. Solid-State Circuits*, vol. 35, no. 7, pp. 1039–1045, Jul. 2000.
[56] B. Razavi, *RF Microelectronics*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1998.
[57] S.-J. Lee, B. Kim, and K. Lee, "A novel high-speed ring oscillator for multiphase clock generation using negative skewed delay scheme," *IEEE J. Solid-State Circuits*, vol. 32, no. 2, pp. 289–291, Feb. 1997.
[58] L. Sun and T. A. Kwasniewski, "A 1.25-GHz 0.35-μm monolithic CMOS PLL based on a multiphase ring oscillator," *IEEE J. Solid-State Circuits*, vol. 36, no. 6, pp. 910–916, Jun. 2001.
[59] E. Hegazi, H. Sjoland, and A. A. Abidi, "A filtering technique to lower LC oscillator phase noise," *IEEE J. Solid-State Circuits*, vol. 36, no. 12, pp. 1921–1930, Dec. 2001.
[60] O. Eliezer, B. Staszewski, and S. Vemulapalli, "Digitally controlled oscillator in a 65 nm GSM/EDGE transceiver with built-in compensation for capacitor mismatches," in *IEEE Radio Frequency Integrated Circuits Symp.*, 2011, pp. 5–7.
[61] O. Eliezer, B. Staszewski, J. Mehta, F. Jabbar, and I. Bashir, "Accurate self-characterization of mismatches in a capacitor array of a digitally-controlled oscillator," in *Proc. IEEE Dallas Circuits and Systems Workshop*, 2010, pp. 17–18.
[62] X. Gao, E. Klumperink, P. Geraedts, and B. Nauta, "Jitter analysis and a benchmarking figure-of-merit for phase-locked loops," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 56, no. 2, pp. 117–121, Feb. 2009.

Colin Weltin-Wu (M'15) received the B.S. degree in mathematics and the M.E. degree in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, and the Ph.D. degree from Columbia University, New York, NY, USA.

He is currently with Analog Devices, San Jose, CA, USA. He spent 15 months starting in 2014 collaborating with Dr. Ian Galton at the University of California, San Diego, CA, USA.

Guobi Zhao received the B.Sc. degree in electrical engineering from the University of Electronic Science and Technology of China, Chengdu, China, in 2012, and the M.S. degree in electrical engineering from the University of California, San Diego, CA, USA, in 2014.

He is currently with Spreadtrum, Shanghai, China, as a design engineer working on PLL and data converters.

Ian Galton (M'92–SM'09–F'11) received the Sc.B. degree from Brown University, Providence, RI, USA, in 1984, and the M.S. and Ph.D. degrees from the California Institute of Technology, Pasadena, CA, USA, in 1989 and 1992, respectively, all in electrical engineering.

Since 1996, he has been a Professor of electrical engineering at the University of California, San Diego, CA, USA, where he teaches and conducts research in the field of mixed-signal integrated circuits and systems for communications. Prior to 1996, he was with the University of California, Irvine, CA, USA, and prior to 1989 he was with Acuson and Mead Data Central. His research involves the invention, analysis, and integrated circuit implementation of critical communication system blocks such as data converters and phase-locked loops.

Dr. Galton is a Fellow of the IEEE. He has served on a corporate Board of Directors, on several corporate Technical Advisory Boards, as the Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II: ANALOG AND DIGITAL SIGNAL PROCESSING, as a member of the IEEE Solid-State Circuits Society Administrative Committee, as a member of the IEEE Circuits and Systems Society Board of Governors, as a member of the IEEE International Solid-State Circuits Conference Technical Program Committee, and as a member of the IEEE Solid-State Circuits Society Distinguished Lecturer Program.