Spectral Breathing and its Mitigation in Digital Fractional-*N* PLLs

Enrique Alvarez-Fontecilla, Eslam Helal, Amr I. Eissa, and Ian Galton, *Fellow, IEEE*

***Abstract*—Although digital phase-locked loops (PLLs) offer several advantages over their analog counterparts, they suffer from a major disadvantage that is rarely mentioned in published papers. The disadvantage, known as spectral breathing, is caused by component mismatches among the frequency control elements within a PLL's digitally-controlled oscillator (DCO). The mismatches introduce DCO frequency modulation nonlinearity which fluctuates and, therefore, causes erratic variations in the PLL's measured phase noise spectrum as the DCO's center frequency drifts. The phenomenon is called spectral breathing because the measured phase noise spectrum tends to slowly swell and contract over time as if taking breaths of air. During these breaths, the PLL's phase noise often becomes severely degraded. This paper presents an experimental demonstration of the spectral breathing phenomenon and its solution in a digital fractional-*N* PLL. The demonstrated solution is a multi-rate dynamic element matching technique and a mismatch-noise cancellation technique that together eliminate spectral breathing.**

***Index Terms*—Digital calibration, digitally-controlled oscillator (DCO), digital phase-locked loop (PLL), frequency control element (FCE), frequency synthesis, inter-symbol interference (ISI), multi-rate dynamic element matching (MR-DEM), mismatch-noise cancellation (MNC), spectral breathing.**

I. Introduction

DIGITAL fractional-*N* phase-locked loops (PLLs) offer several advantages over analog PLLs, such as lower loop filter circuit area and better compatibility with highly-scaled CMOS IC technology [1]–[20]. However, unlike their analog counterparts, many digital PLLs suffer from a problem called *spectral breathing* which can significantly degrade phase noise performance.

Spectral breathing is the result of nonlinear frequency modulation from inevitable mismatches among the frequency control elements (FCEs) in a PLL's digitally-controlled oscillator (DCO) [21]. Flicker noise, temperature variations, and supply voltage variations cause the DCO's free-running frequency to drift over time, so different FCEs are exercised as the PLL adjusts the DCO's input sequence to compensate for the drift. This causes the DCO's frequency modulation nonlinearity to vary and, therefore, the PLL's measured phase noise spectrum to swell up from time to time as if taking breaths of air.

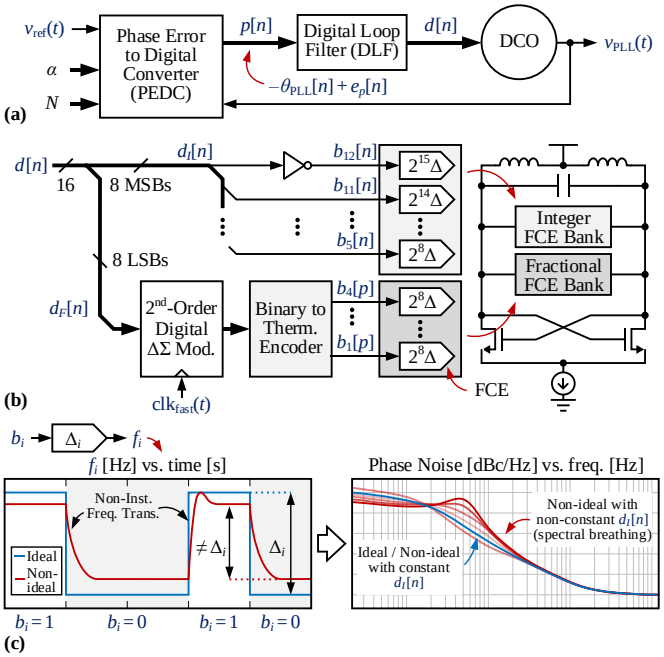

Fig. 1. (a) General form of a digital fractional-*N* PLL (b) conventional DCO control technique, and (c) illustration of the effects of non-ideal FCE behavior on the PLL's phase noise spectrum.

The effect is particularly significant in the large percentage of digital PLLs that use a digital delta-sigma (ΔΣ) modulator to oversample the fractional part of the DCO input. In such PLLs, the ΔΣ modulator drives a bank of FCEs called the *fractional FCE bank* and the integer part of the DCO input drives a separate bank of FCEs called the *integer FCE bank*. The fractional FCE bank's FCEs are exercised many times per reference period because they are driven by the oversampling ΔΣ modulator. In contrast, the integer FCE bank's FCEs are exercised at most once per reference period because they are driven directly by the integer part of the DCO's input sequence. Accordingly, error introduced by the integer FCE bank as a result of FCE mismatches is the main problem; it is spread over a much smaller Nyquist band than the corresponding error introduced by the fractional FCE bank, so much less of it is suppressed by the DCO's lowpass frequency-to-phase filtering operation.

Perhaps because the problem is not visible when measurements are restricted to time periods during which the integer FCE bank input bits do not change, it is seldom mentioned in the published literature. Nevertheless, in practice the integer FCE bank input bits do change, at least over time periods of several seconds, because of the DCO's free-running frequency

Manuscript received January XX, 2021, Revised XX, 2021. This work was supported by the National Science Foundation under Award 1617545. This paper was recommended by Associate Editor XXXXXXXXX. *(Corresponding author: Enrique Alvarez-Fontecilla)*

E. Alvarez-Fontecilla is with Analog Devices Inc, San Diego, CA 92128 USA (e-mail: enrique.alvarezfontecilla@analog.com).

E. Helal, A. I. Eissa and I. Galton are with the Electrical and Computer Engineering Department, University of California, San Diego, La Jolla, CA 92092-0407 USA.

drift, so the problem is significant in applications.

To the knowledge of the authors, the phenomenon was first reported and explained, although not yet named, in [2]. The PLL presented in [2] uses the DCO frequency control method described above, which was already well-known and widely-used at the time, so the authors of [2] were surprised to find that FCE mismatches caused the PLL's measured phase noise spectrum to vary significantly over time.

The reason the frequency control method described above is so widely used in digital PLLs is that it is very power-efficient because only the fractional part of the DCO input sequence is oversampled. Unfortunately, this very feature is what exacerbates the spectral breathing problem. To date, only three solutions that mitigate spectral breathing yet retain this feature have been proposed [21]. An offline calibration technique is proposed in [22] and [23] to compensate for FCE mismatches, but the technique requires several minutes to complete. A digital integer-boundary avoider circuit prior to the DCO is proposed in [3], yet the technique is only effective in cases where the integer FCE bank input bits change relatively infrequently. To the authors' knowledge, the only published comprehensive solution is the combination of a multi-rate dynamic element matching (MR-DEM) technique and a mismatch-noise cancelation (MNC) technique [21].

This paper presents a fractional-*N* digital PLL IC that incorporates the MR-DEM and MNC techniques to avoid spectral breathing. In addition to providing the first experimental demonstration of the techniques proposed in [21], it presents implementation details, refinements, and practical observations that are not presented in [21].

II. Background Information

A. *Conventional DCO Frequency Control Techniques*

Most fractional-*N* digital PLLs have the general structure shown in Fig. 1(a), which consists of a phase-error-to-digital converter (PEDC), a digital loop filter (DLF), and a DCO. The PLL's input, $v_{ref}(t)$, is generated by a reference oscillator not shown in the figure and the PLL's output, $v_{PLL}(t)$, ideally is periodic with frequency $f_{PLL} = (N + \alpha)f_{ref}$, where f_{ref} is the reference oscillator frequency, N is a positive integer, and α is a fractional value. The PEDC generates an f_{ref}-rate sequence of the form $-\theta_{PLL}[n] + e_p[n]$, where $\theta_{PLL}[n]$ is the PLL's phase error and $e_p[n]$ represents the combined effect of all other errors. The PEDC output is filtered by the DLF, and the DLF output, $d[n]$, controls the DCO frequency.

The DCO's instantaneous frequency, $f_{DCO}(t)$, can be written as a fixed offset frequency plus $f_{tune}(t)$, where $f_{tune}(t)$ depends on the states of the DCO's FCEs. Ideally, $f_{tune}(t) = d[n_t]$, where $n_t = n$ over the nth period of the f_{ref}-rate clock that updates $d[n]$, so n_t is a continuous-time function that takes on values $n = 0, 1, 2, 3, \ldots$ as t increases. If $d[n]$ is represented as a b-bit two's complement code sequence and its least-significant bit (LSB) represents a frequency step of Δ, then, $d[n_t] = (-2^{b-1}d_{b-1}[n_t] + 2^{b-2}d_{b-2}[n_t] + \ldots + 2^0 d_0[n_t])\Delta$, where $d_i[n_t]$ is the ith bit of the $d[n_t]$ code for $i = 0, 1, \ldots, b-1$. In principle, the DCO could be implemented with b FCEs, where the ith FCE increases or decreases $f_{DCO}(t)$ by $2^i\Delta$ whenever the FCE's input bit changes from 0 to 1 or 1 to 0, respectively. In this case, the input to the ith FCE would be $d_i[n_t]$ for $i = 0, 1, \ldots, b-2$, and $1 - d_i[n_t]$ for $i = b-1$. Unfortunately, in most PLLs this would require FCEs with impractically small frequency steps.

Fig. 1(b) shows an example configuration of a DCO control technique that is widely used to circumvent this problem [1]. In this example, the minimum practical FCE frequency step is $\Delta_{min} = 2^8\Delta$, and $d[n]$ is decomposed into an integer part, $d_I[n]$, that takes on values that are multiples of Δ_{min}, and a fractional part, $d_F[n]$, that takes on values in the range $\{0, \Delta, 2\Delta, \ldots, \Delta_{min} - \Delta\}$. The $d_I[n]$ sequence drives an *integer FCE bank* directly. The $d_F[n]$ sequence is oversampled at a rate of $f_{fast} > f_{ref}$ and requantized by a digital $\Delta\Sigma$ modulator. The $\Delta\Sigma$ modulator output is converted to a thermometer code that drives a *fractional FCE bank* which consists of four FCEs, each with a frequency step of Δ_{min}. Thus, $f_{tune}(t)$ is equal to $d[n_t]$ plus f_{fast}-rate highpass-shaped $\Delta\Sigma$ modulator quantization error that is lowpass filtered by the DCO.

Ideally, the contribution to $f_{DCO}(t)$ from the ith FCE instantaneously increases or decreases by Δ_i when the FCE's input bit changes from 0 to 1 or 1 to 0, respectively, where Δ_i is the FCE's nominal frequency step. In practice, non-ideal circuit behavior causes the FCE's frequency transitions to be non-instantaneous, and component mismatches cause its frequency step to deviate somewhat from Δ_i as illustrated in Fig. 1(c). These nonidealities introduce input-code-dependent DCO frequency modulation nonlinearity. As illustrated in Fig. 1(c), this causes the PLL's phase noise spectrum to vary over time as the DCO's free-running frequency, and, hence, $d_I[n]$, drift [21].

In most low-noise PLLs that generate unmodulated output tones, the short-term variation of $d[n]$ is much less than Δ_{min}. In such cases, the PLL output frequency can be tuned such that $d[n]$ is far enough from an integer boundary that $d_I[n]$ initially remains constant. To the extent that this condition persists over time, the integer FCE bank does not introduce dynamic error. However, flicker noise, temperature variations, and supply voltage variations inevitably cause the DCO's free-running frequency to drift, and the PLL responds by adjusting $d[n]$ to compensate for the drift. Eventually, this causes $d[n]$ to get close enough to an integer boundary that $d_I[n]$ begins to toggle, thereby causing the spectral breathing problem. Typically, such spectral breathing events occur at time intervals ranging from a few seconds to a few minutes.

Even if the DCO's free-running frequency drift could somehow be avoided, preventing the integer FCE bank from introducing dynamic error by avoiding PLL output frequencies for which $d_I[n]$ toggles would not be practical in most applications because it would arbitrarily prohibit the use of several PLL output frequency bands. Moreover, in PLLs designed to generate frequency or phase modulated output waveforms, the short term variation of $d[n]$ typically exceeds Δ_{min}, in which case the integer FCE bank continuously introduces dynamic error and severely degrades the PLL's phase noise.

One solution is to apply DEM clocked at the $\Delta\Sigma$ modulator output rate to all the DCO's FCEs. This causes all error from FCE mismatches to be free of nonlinear distortion and shaped to high-enough frequencies that it is largely suppressed by the DCO. Unfortunately, the solution is prohibitively power hungry because of the high sample-rate, and having so many rap-

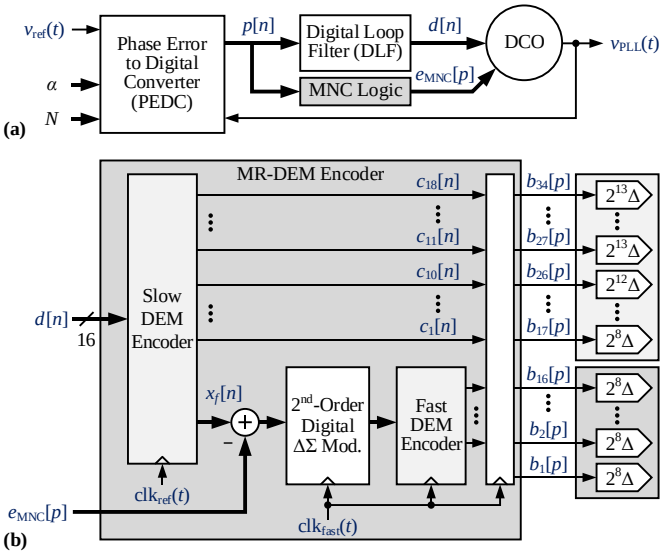

Fig. 2. (a) Block diagram of a digital fractional-*N* PLL with the MR-DEM and MNC techniques, and (b) MR-DEM technique details.

idly toggling digital lines routed in close proximity to the DCO's tank tends to couple noise and spurious tones into the DCO output.

A somewhat lower-power, but only partial solution is to have a single FCE bank driven by the output of an oversampling ΔΣ modulator with input $d[n]$ rather than just $d_F[n]$ [16]. While this option does not cause error from FCE mismatches to be free of nonlinear distortion or shaped to high frequencies, it does spread the error over a larger Nyquist band so that more of it gets suppressed by the DCO. Hence, it reduces, but does not eliminate, the spectral breathing problem, and it does not necessarily avoid the problem of having a large number of rapidly toggling digital lines routed in close proximity to the DCO's tank.

The latter problem can be avoided by implementing the DCO as an oversampling ΔΣ digital-to-analog converter (DAC) followed by a voltage controlled oscillator (VCO) [9], [15], [17]–[20]. If the ΔΣ DAC implements DEM, then spectral breathing is avoided, but the problem of high power consumption remains. Without DEM, as long as the ΔΣ DAC oversamples $d[n]$ and not just $d_F[n]$, then the spectral breathing problem is mitigated, but still not eliminated, for the reasons described above.

B. *MR-DEM and MNC Techniques*

Fig. 2(a) shows a high-level diagram of a digital fractional-*N* PLL that includes the MR-DEM and MNC techniques. The system is similar to that shown in Fig. 1(a), except for the addition of two digital blocks: an MR-DEM encoder that is built into the DCO and an MNC logic block.

The MR-DEM encoder is based on the same principle as mismatch-shaping DEM encoders [24]. The idea is to shuffle the error introduced by FCE mismatches, so that $f_{tune}(t)$ equals a scaled version of $d[n_t]$ plus additive highpass-shaped error that depends on pseudo-random digital sequences that are known to the system because they are generated within the MR-DEM encoder. This additive error has both f_{ref}-rate and f_{fast}-rate components. Most of the f_{ref}-rate error component in $f_{tune}(t)$ is canceled by the MNC technique, which applies a least-

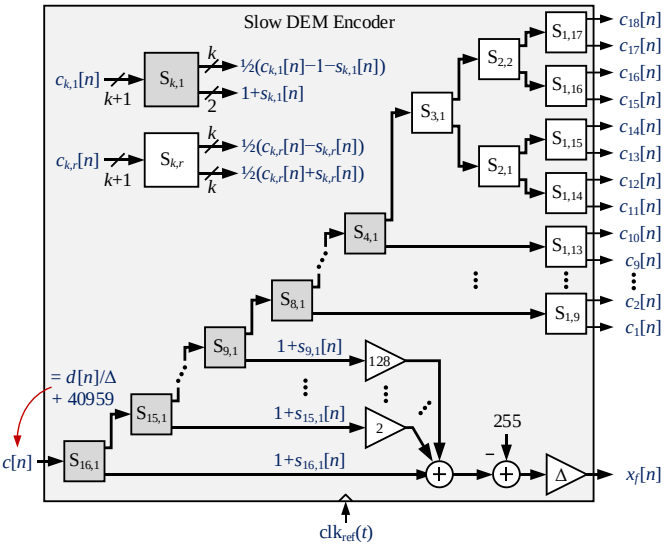

Fig. 3. Slow DEM encoder signal processing.

mean-square (LMS)-like algorithm to compute digital coefficients with which it forms a correction sequence, $e_{MNC}[p]$, that is injected into the MR-DEM encoder. The f_{fast}-rate error component is not canceled by the MNC technique, but its high sample-rate in conjunction with its highpass spectral shape ensures that most of its contribution to the DCO's phase noise gets suppressed by the DCO's first-order lowpass frequency-modulation-to-phase-noise transfer function.

Fig. 2(b) shows the details of the MR-DEM technique for an example case where $d[n]$ has 16 bits. It consists of an MR-DEM encoder that comprises a slow DEM encoder, a second-order digital ΔΣ modulator, and a fast DEM encoder. The slow DEM encoder is a modified version of a conventional segmented DEM encoder, the details of which are described shortly, and the fast DEM encoder is a conventional non-segmented DEM encoder. The slow DEM encoder is clocked by the f_{ref}-rate clock signal, $\text{clk}_{ref}(t)$, whereas the MR-DEM encoder's fractional path, which consists of the digital ΔΣ modulator and the fast DEM encoder, is clocked by the f_{fast}-rate clock signal, $\text{clk}_{fast}(t)$.

The signal processing details of the slow DEM encoder are shown in Fig. 3, where the Δ multiplication prior to the $x_f[n]$ output denotes that the LSB of $x_f[n]$ represents a frequency step of Δ. The slow DEM encoder consists of 25 digital switching blocks (SBs), labeled $S_{k,r}$ for k = 1, 2, …, 16 and r = 1, 2, …, 17. The shaded SBs are called segmenting SBs, whereas the remaining SBs are called non-segmenting SBs. The functional details of each SB type are also shown in Fig. 3, where $c_{k,r}[n]$ is the input sequence of $S_{k,r}$. The outputs of each segmenting SB are $½(c_{k,1}[n] - 1 - s_{k,1}[n])$ and $1 + s_{k,1}[n]$, where $s_{k,1}[n]$, called a switching sequence, is 0 when $c_{k,1}[n]$ is odd and ±1 otherwise. The outputs of each non-segmenting SB are $½(c_{k,r}[n] - s_{k,r}[n])$ and $½(c_{k,r}[n] + s_{k,r}[n])$, where $s_{k,r}[n]$ is 0 when $c_{k,r}[n]$ is even and ±1 otherwise. Each switching sequence is zero-mean and has a highpass-shaped power spectral density that peaks at $f_{ref}/2$.

As explained in [21], in the absence of the MNC technique the MR-DEM encoder would cause

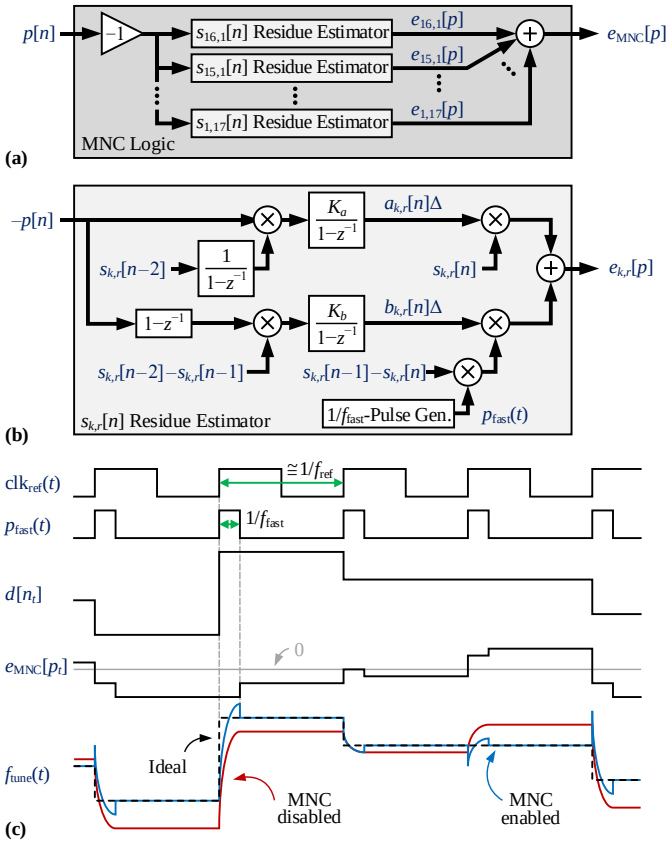

Fig. 4. (a) MNC logic signal processing, (b) $s_{k,r}[n]$ residue estimator signal processing, and (c) illustration of the MNC technique operation where $p_t = p$ over the pth period of $\mathrm{clk}_{\mathrm{fast}}(t)$.

$$f_{\mathrm{tune}}(t) \cong d[n_t] + \Delta \sum_{k,r} \delta_{k,r} s_{k,r}[n_t] + p_{\mathrm{fast}}(t) \Delta \sum_{k,r} \gamma_{k,r}(t) \left(s_{k,r}[n_t - 1] - s_{k,r}[n_t] \right), \quad (1)$$

where $\delta_{k,r}$ and $\gamma_{k,r}(t)$ are constant and $1/f_{\mathrm{fast}}$-periodic waveforms, respectively, that depend only on FCE errors, and $p_{\mathrm{fast}}(t)$ is a series of unit-amplitude, $1/f_{\mathrm{fast}}$-width pulses that go high whenever n_t changes.[2] As implied by (1), the MR-DEM technique causes $f_{\mathrm{tune}}(t)$ to be a linear function of $d[n_t]$ at the expense of introducing two additive error terms. One of the error terms is caused by FCE static gain errors, i.e., mismatch-induced errors in the FCEs' frequency step-sizes. The other error term is inter-symbol interference (ISI) that results from non-instantaneous rise and fall frequency transients of individual FCEs [21].

The details of the MNC logic are shown in Fig. 4(a). It consists of 25 $s_{k,r}[n]$ residue estimators that each compute a correction sequence corresponding to one of the slow DEM encoder's switching sequences, and an adder that combines these sequences to form $e_{\mathrm{MNC}}[p]$. As shown in Fig. 4(b), each $s_{k,r}[n]$ residue estimator comprises two branches, one to compute the correction sequence associated with FCE static gain error, and another to compute the correction sequence associated with ISI error, hereafter referred to as the static-error and ISI-error correction branches, respectively.

[2] The summation indices in (1) indicate the summation over all k and r values corresponding to the SBs within the slow DEM encoder.

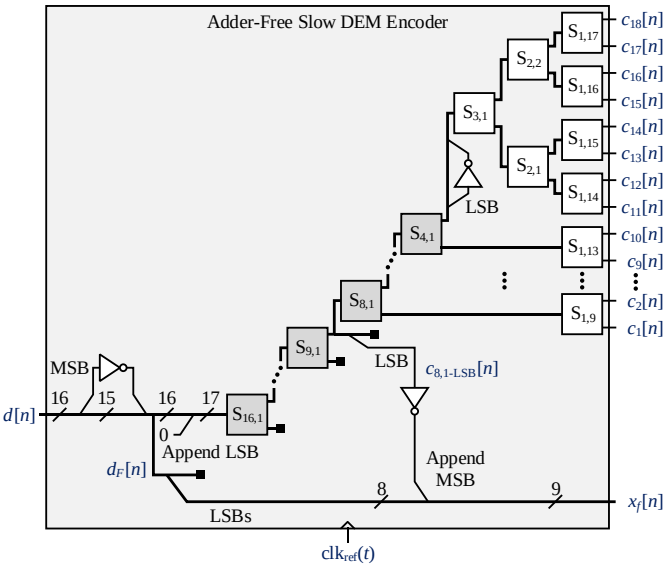

Fig. 5. Adder-free slow DEM encoder implementation.

As indicated in Fig. 4, the MNC logic block's output is

$$e_{\mathrm{MNC}}[p] = \Delta \sum_{k,r} a_{k,r}[n] s_{k,r}[n] + p_{\mathrm{fast}}(t) \Delta \sum_{k,r} b_{k,r}[n] \left(s_{k,r}[n-1] - s_{k,r}[n] \right), \quad (2)$$

where $a_{k,r}[n]$ and $b_{k,r}[n]$ are measures called MNC coefficients that correspond to $\delta_{k,r}$ and $\gamma_{k,r}(t)$, respectively. As explained in [21], the MNC coefficients converge such that $e_{\mathrm{MNC}}[p]$ is a sampled measure of the additive error terms in (1). As illustrated conceptually in Fig. 4(c), when injected into the DCO as shown in Fig. 2(b), $e_{\mathrm{MNC}}[p]$ largely prevents these terms from contributing to the DCO's phase noise.

III. Implementation Details

The MR-DEM and MNC techniques were implemented as modifications to the 6.5 GHz digital fractional-N PLL presented in [4]. The PEDC is based on a second-order ΔΣ frequency-to-digital converter (FDC). The ΔΣ FDC is based on dual-mode ring oscillator (DMRO) and it incorporates the all-digital enhancements described in [25] in addition to a time amplifier (TA) prior to the phase-frequency detector (PFD). The DLF consists of a loop gain multiplier, three single-pole IIR stages, and a proportional-integral stage. As the details of the underlying PLL are explained in [4], only the additional implementation details relevant to the MR-DEM and MNC techniques are presented here.

A. *MR-DEM Encoder*

The high-level details of the implemented MR-DEM encoder are identical to those shown in Fig. 2(b). The second-order digital ΔΣ modulator is implemented as an error feedback structure to reduce its hardware complexity [26]. The fast DEM encoder is implemented as a 4-layer tree of non-segmenting SBs with first-order highpass-shaped switching sequences [27], [28]. The slow DEM encoder is implemented as shown in Fig. 5. The SBs within the slow and fast DEM

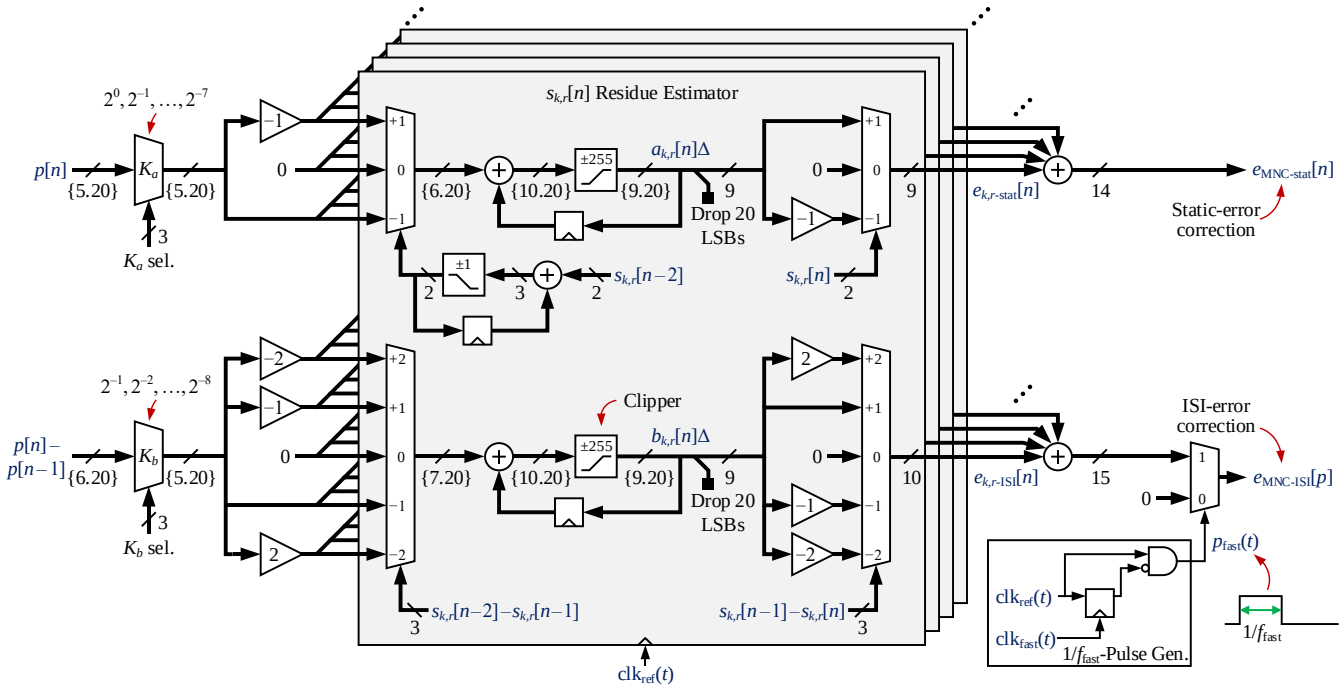

Fig. 6. MNC logic bit-level implementation.

encoders are identical to the adder-free SBs described in [29], except for a modification made in the slow DEM encoder's SBs for them to accommodate both white and first-order highpass-shaped switching sequences.

It follows from Fig. 3 that the slow DEM encoder bottom output, $x_f[n]$, is given by

$$\Delta\sum_{k=9}^{16} 2^{16-k}\, s_{k,1}[n], \tag{3}$$

which can be computed by combining the bottom outputs of $S_{16,1}$, $S_{15,1}$, … and $S_{9,1}$ as indicated in Fig. 3. However, in this work $x_f[n]$ is instead computed without using adders to reduce the block's hardware complexity. As explained shortly, the LSB, $c_{8,1\text{-LSB}}[n]$, of the input to $S_{8,1}$, $c_{8,1}[n]$, corresponds to a quantized version of $d_F[n]$, and its quantization error is proportional to (3). The proposed slow DEM encoder architecture takes advantage of this property to generate $x_f[n]$ by simply combining the bits of the $d_F[n]$ and $c_{8,1\text{-LSB}}[n]$ sequences as indicated in Fig. 5.

Fig. 5 implies that

$$x_f[n] = -2^8\left(1-\frac{c_{8,1\text{-LSB}}[n]}{\Delta}\right)\Delta + d_F[n], \tag{4}$$

and the SB signal processing operations shown in Fig. 3 imply that

$$c_{8,1}[n] = \frac{1}{2^8}\left(c[n]+1-2^8-\sum_{k=9}^{16} 2^{16-k}\, s_{k,1}[n]\right)\Delta. \tag{5}$$

As explained in [29], the segmenting SBs use negative-extra-LSB encoding, so $c_{8,1\text{-LSB}}[n]$ has a negative weight. This with the segmenting SB details presented in [29] implies that $c_{8,1\text{-LSB}}[n]$ is given by minus the right side of (5) with $c[n]$ replaced by $d_F[n]/\Delta$, i.e.,

$$c_{8,1\text{-LSB}}[n] = -\frac{1}{2^8}\left(\frac{d_F[n]}{\Delta}+1-2^8-\sum_{k=9}^{16} 2^{16-k}\, s_{k,1}[n]\right)\Delta. \tag{6}$$

Substituting (6) into (4) yields (3) plus a $-\Delta$ offset. This offset is not a problem in practice because it is suppressed by a zero-frequency zero in the PLL's transfer function.

B. *MNC Logic*

The MNC logic implementation details are shown in Fig. 6. Each variable is represented in fixed-point, two's complement format, and its number of bits is specified via the notation $\{b_I.\, b_F\}$, where b_I and b_F are the numbers of integer and fractional bits, respectively. The braces and b_F are omitted in cases where $b_F = 0$.

Each $s_{k,r}[n]$ residue estimator computes two sequences, $e_{k,r\text{-stat}}[n]$ and $e_{k,r\text{-ISI}}[n]$. The $e_{k,r\text{-stat}}[n]$ sequences are combined to form $e_{\text{MNC-stat}}[n]$, which corresponds to the first summation in (2), i.e., the part of $e_{\text{MNC}}[p]$ associated with FCE static gain errors. The $e_{k,r\text{-ISI}}[n]$ sequences are combined and then multiplied by the unit-amplitude pulse sequence, $p_{\text{fast}}(t)$, to form $e_{\text{MNC-ISI}}[p]$, which corresponds to the second summation in (2), i.e., the part of $e_{\text{MNC}}[p]$ associated with FCE ISI errors. Although not shown in Fig. 6, $e_{\text{MNC-stat}}[n]$ and $e_{\text{MNC-ISI}}[p]$ are combined at the $\Delta\Sigma$ modulator's input in Fig. 2(b) to form $e_{\text{MNC}}[p]$.

As explained in [28], each $s_{k,r}[n]$ is a concatenation of sequences of the form 1, 0, …, 0, –1, 0, …, 0 or –1, 0, …, 0, 1, 0, …, 0. Thus, $|s_{k,r}[n]| \le 1$ and $|s_{k,r}[n] - s_{k,r}[n-1]| \le 2$ for all n, and the running sum of $s_{k,r}[n]$ never exceeds 1 nor –1. However, after startup, the finite state machines that generate the switching sequences do not necessarily start at the beginning of their respective cycles, which could cause the magnitudes of the running sums of some switching sequences to exceed 1. This issue is avoided by the inclusion of a ±1 clipper within each $s_{k,r}[n]$ accumulator as shown in Fig. 6.

Each MNC coefficient within each $s_{k,r}[n]$ residue estimator is the output of a 29-bit clipping accumulator. The 20 LSBs of

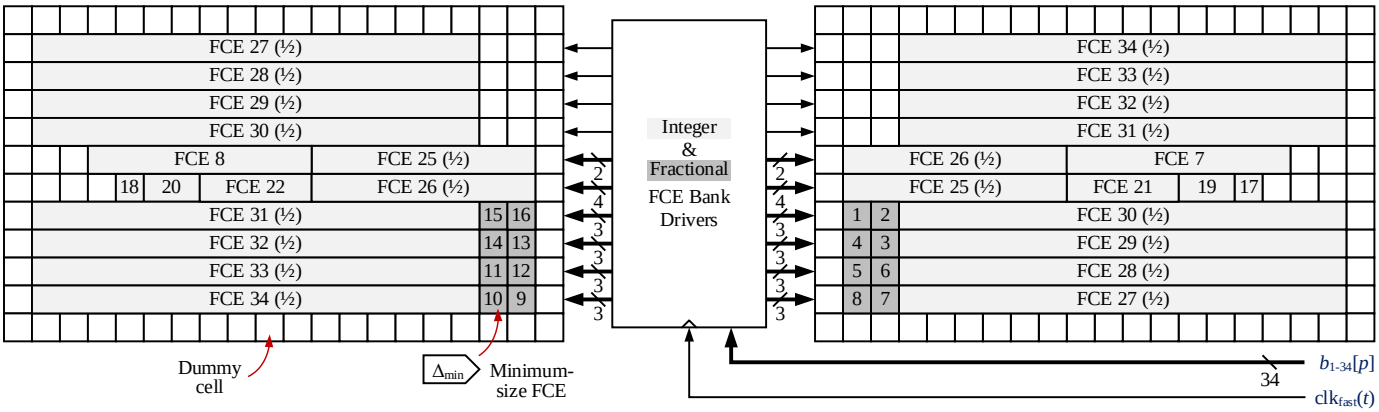

Fig. 7. Integer and fractional FCE banks layout.

the coefficients are dropped prior to their respective multiplication by $s_{k,r}[n]$ and $s_{k,r}[n-1] - s_{k,r}[n]$, which reduces power consumption at the expense of reducing the accuracy with which $e_{MNC}[p]$ cancels the additive error terms in (1). The number of bits to drop was determined with the aid of simulations performed by the authors using a bit-accurate, event-driven, C-language, custom PLL simulator such that the contribution to the PLL's phase noise from the residual error that is left after MNC is applied is negligible.

The purpose of the clipping operations in the $s_{k,r}[n]$ residue estimators is to reduce the maximum MNC coefficient convergence time by preventing the accumulator magnitudes from becoming unnecessarily large prior to convergence. Provided the clipping bounds are larger than the magnitudes of the converged MNC coefficients, they have no effect once the MNC coefficients converge. The clipper bounds of ±255 were chosen with the aid of behavioral simulations that considered pessimistic FCE mismatches. None of the MNC coefficient magnitudes were expected to exceed 128 after convergence, so the bounds of ±255 provide considerable margin.

The K_a and K_b gains are restricted to powers of 2 so the implementation of their respective multipliers only involves bus-shifting. Consequently, as shown in Fig. 6, the MNC logic requires no actual digital multipliers.

C. *DCO FCE Banks*

The DCO consists of a single-turn center-tapped inductor, a cross-coupled NMOS pair, a tail resonant tank [30], a triode-MOS tail source, and integer and fractional FCE banks. The implemented FCEs are of the type presented in [2], and the minimum-size FCE has an equivalent frequency step of Δ_{min} = 160 kHz at 6.5 GHz. The integer FCE bank comprises eight $32\times\Delta_{min}$ FCEs and five pairs of 16×, 8×, 4×, 2× and 1× Δ_{min} FCEs, whereas the fractional FCE bank comprises sixteen Δ_{min} FCEs. All FCEs are implemented by connecting one or more Δ_{min} FCEs in parallel.

Both FCE banks are laid out as illustrated in Fig. 7, where the FCEs are indexed according to their respective control bits. Each FCE is driven by a flip-flop clocked by $clk_{fast}(t)$, followed by a buffer. The size of each buffer is scaled according to the number of parallel Δ_{min} FCEs it drives so as to minimize the systematic mismatches among the FCE frequency transients, and, hence, lower the error contribution from ISI. Behavioral simulations performed by the authors in which estimates of the FCE frequency transients were implemented

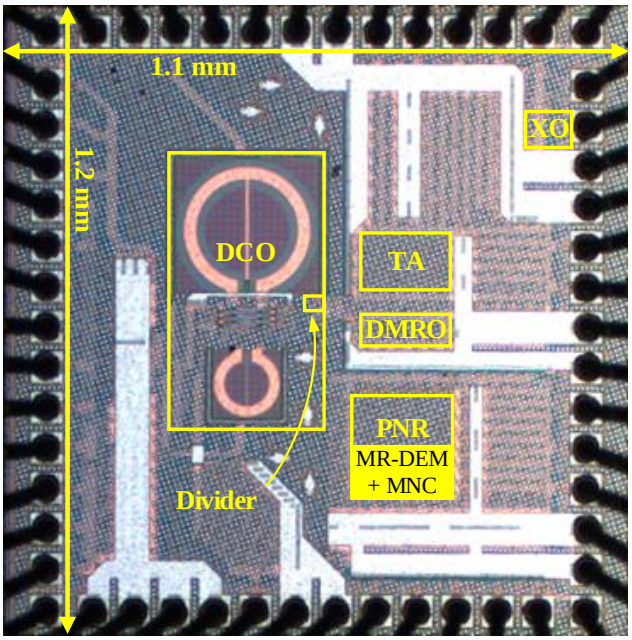

Fig. 8. Die micrograph.

suggest that for the specific FCE architecture used, the contribution to the PLL's phase noise from the MR-DEM technique's ISI error component is much less significant than that of the static gain error component. This observation was experimentally validated, as demonstrated in Section IV.

As shown in Fig. 7, the ten largest FCEs of the integer FCE bank, FCEs 25 to 34, are split into two halves each, which are laid out in a common-centroid fashion to avoid the FCE mismatches from being exacerbated. As explained in Section II-B, the MNC technique cancels much of the error that arises from FCE mismatches, but the larger the FCE mismatches, the larger the required dynamic range of $e_{MNC}[p]$ and the larger the resulting output dynamic range of the MR-DEM encoder's fractional path. A larger fractional path output dynamic range is undesirable in practice as it increases the MR-DEM encoder's power consumption, as well as the number of control lines that need to be routed from the DCO control to the FCE banks' drivers. Therefore, care was taken with the layout to minimize FCE mismatches.

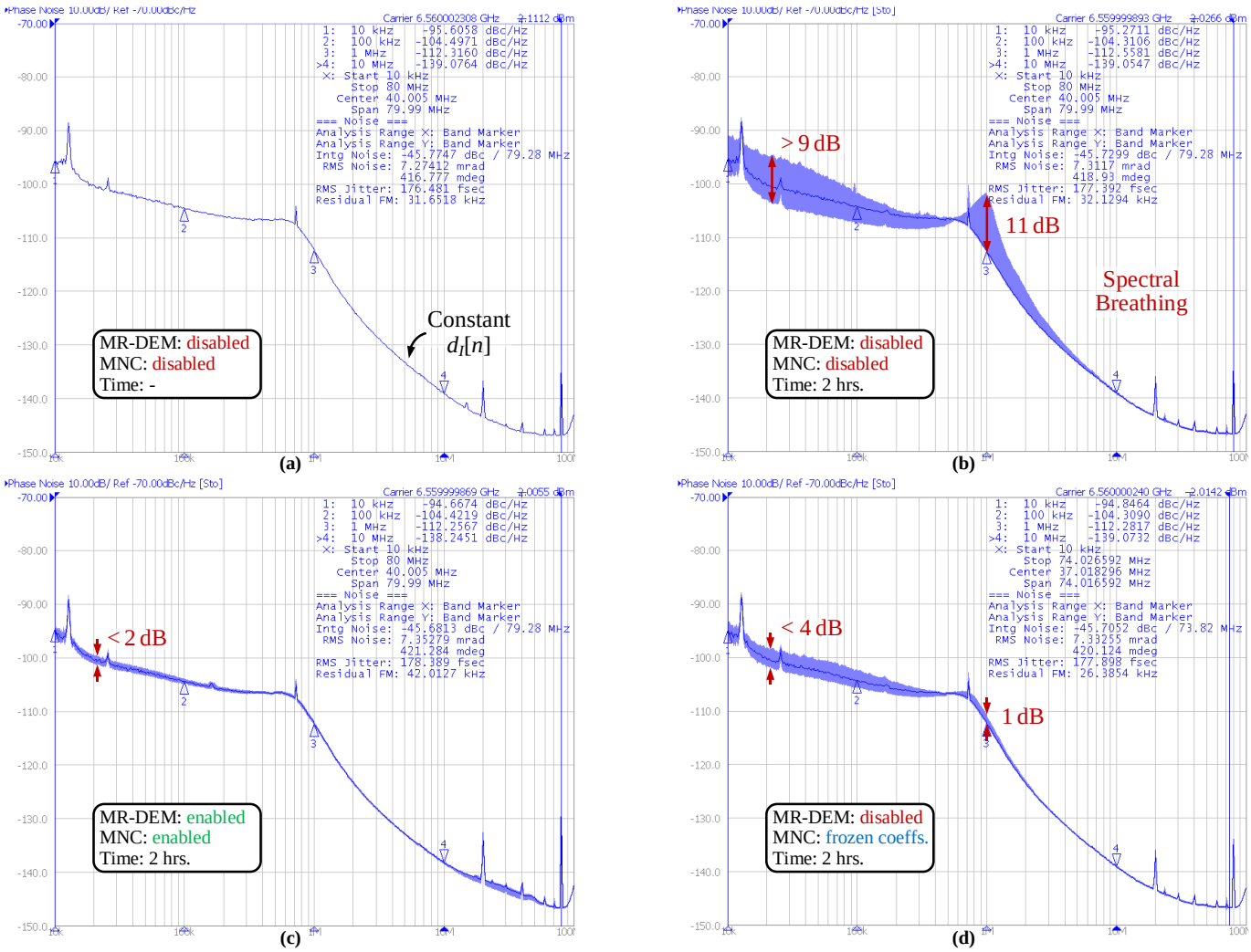

Fig. 9. Measured PLL phase noise at f_{PLL} = 6.56 GHz for (a) conventional DCO control technique with constant $d_l[n]$, (b) conventional DCO control technique and persistence enabled for 2 hrs. (c) MR-DEM and MNC enabled and persistence enabled for 2 hrs., and (d) MR-DEM disabled, MNC enabled with coefficients frozen after convergence, and persistence enabled for 2 hrs.

IV. Measurement Results

A die photograph of the prototype IC is shown in Fig. 8. The IC consists of the digital fractional-*N* PLL as well as a serial peripheral interface (SPI) and additional circuitry used for testing. The IC was fabricated in the Global-Foundries 22 nm CMOS 22FDX technology.

The IC's place-and-route (PNR) digital block has a 0.8 V power supply and is clocked by an f_{fast}-rate clock, where $f_{fast} = f_{PLL}/8$, which is synchronous to the DCO output. However, most of the PNR digital block's sub-blocks are clocked by f_{ref} = 80 MHz clocks that are derived from the f_{fast}-rate clock. The maximum value of f_{fast} is 830 MHz, although the PNR digital block was designed to run at a clock-rate as high as 1 GHz to provide design margin. The PNR digital block's area is 0.0482 mm^2, half of which corresponds to the circuitry associated with the MR-DEM and MNC techniques.

As shown in Fig. 7, the FCE drivers are also clocked by the f_{fast}-rate clock. The integer bank's FCEs are only updated once per reference period, so their drivers could have been clocked by the reference signal, but this was not done because it would have risked inducing fractional spurs by injecting reference signal interference into the DCO.

Fig. 9 shows the measured PLL phase noise at f_{PLL} = 6.56 GHz for a PLL bandwidth of 900 kHz under various conditions. The signal source analyzer's averaging option was set to 32 when taking each measurement result shown in the figure. In Fig. 9(a), the MR-DEM technique is disabled, i.e., it is configured to operate as a conventional DCO control technique,[3] the MNC technique is disabled, and the measurement was taken over a time period wherein $d_l[n]$ did not vary. Thus, the phase noise profile shown in Fig. 9(a) corresponds to a standard phase noise profile, as reported in most published papers that report digital PLL results. Fig. 9(b) shows the measured PLL phase noise for the same conditions of Fig. 9(a), except that the measurement time duration was increased and the instrument's persistence option was enabled. The PLL was left running for two hours for the measurement, and $d[n]$ was regularly monitored through the SPI to check for integer-

[3] Specifically, the slow DEM encoder's randomization is disabled, in which case the slow DEM encoder behaves as a conventional encoder. The digital ΔΣ modulator and fast DEM encoder were enabled for all the measurements reported in this paper.

boundary crossings. As shown in Fig. 9(b), the PLL phase noise varies significantly over time, such that the spot phase noises at 20 kHz and 1 MHz offset frequencies vary by 9 dB and 11 dB, respectively. The $d[n]$ sequence crossed three integer boundaries several times during the measurement.[4] Additional measurements performed by the authors indicate that phase noise profiles almost identical to that shown in Fig. 9(b) can be easily obtained in a few minutes, and sometimes even in a few seconds.

Despite the considerable variations in the phase noise profile shown in Fig. 9(b), the total integrated jitter, σ_{TJ}, of the PLL did not vary by more than 10 fs_{rms} during the measurement period. This is because in this particular case increases in the spot phase noise around a 1 MHz offset frequency were accompanied by a lower phase noise at offset frequencies below 500 kHz, thus preventing the total integrated jitter from changing significantly. Additional measurements taken by the authors for different PLL parameters suggest that this observation does not hold in general, as σ_{TJ} increases or even decreases by different amounts as the PLL phase noise profile fluctuates over time. Nonetheless, even if σ_{TJ} does not change significantly, as in the case of Fig. 9(b), spectral breathing is still problematic in many applications. For instance, when the PLL is used to generate the local oscillator signal in a wireless transceiver, spectral breathing would cause significant reciprocal mixing error [31].

Fig. 9(c) shows the measured PLL phase noise for the same conditions of Fig. 9(b), except with the MR-DEM and MNC techniques both enabled. In this case, $d[n]$ crossed two integer boundaries several times during the measurement. As demonstrated in Fig. 9(c), the PLL phase noise varies much less with the proposed techniques enabled. Even at offset frequencies lower than 100 kHz, where the phase noise profile is expected to vary somewhat because of the algorithm used by the instrument to compute the spectrum, the variations in Fig. 9(c) are considerably less significant than those in Fig. 9(b) (e.g., less than 2 dB versus more than 9 dB at a 20 kHz offset frequency). As explained below, the ISI-error correction branch of the MNC technique does not perfectly cancel the ISI component of the error at the PLL output. This is reflected in Fig. 9(c) as a slight phase noise increase at offset frequencies above 10 MHz. Additional measurements taken by the authors with both the MR-DEM and MNC techniques enabled suggest that the measured PLL phase noise changes negligibly as $d_l[n]$ varies.

Fig. 9(d) shows the measured PLL phase noise for the same conditions of Fig. 9(c), except that the MR-DEM technique was disabled and the MNC technique's coefficients were frozen after convergence but prior to the measurement. In this case, $d[n]$ crossed two integer boundaries several times during the measurement. The phase noise profile shown in Fig. 9(d) is similar to that shown in Fig. 9(c), except for slightly larger variations at offset frequencies below 200 kHz and at a 1 MHz offset frequency, and the phase noise at offset frequencies above 10 MHz which does not exhibit the excess noise shown in Fig. 9(c). The increase in phase noise variation at low offset frequencies in Fig. 9(d) compared to Fig. 9(c) happens because MNC coefficient noise gets sampled, and, hence, locked in when the MNC coefficients are frozen. Hence, in this configuration, the PLL's performance is expected to be similar to

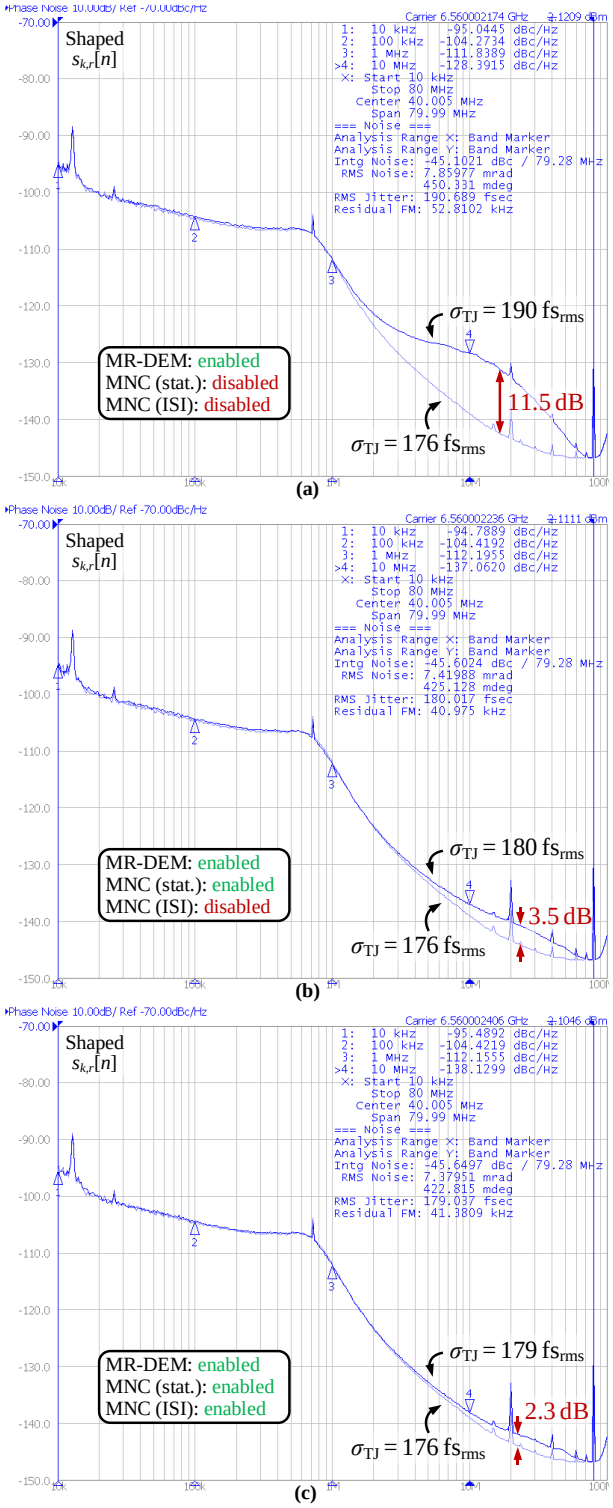

Fig. 10. Measured PLL phase noise at f_{PLL} = 6.56 GHz for (a) MR-DEM enabled and MNC disabled, (b) MR-DEM and MNC (stat. branch only) enabled, and (c) MR-DEM and MNC (both stat. and ISI branches) enabled.

[4] Note that this implies that the DCO's center frequency drifted by more than $2\times\Delta_{min}$ = 320 kHz while the measurement was being taken, so $d_F[n]$, which can take on values from the set {0, Δ, 2Δ, … 255Δ}, rolled over three times.

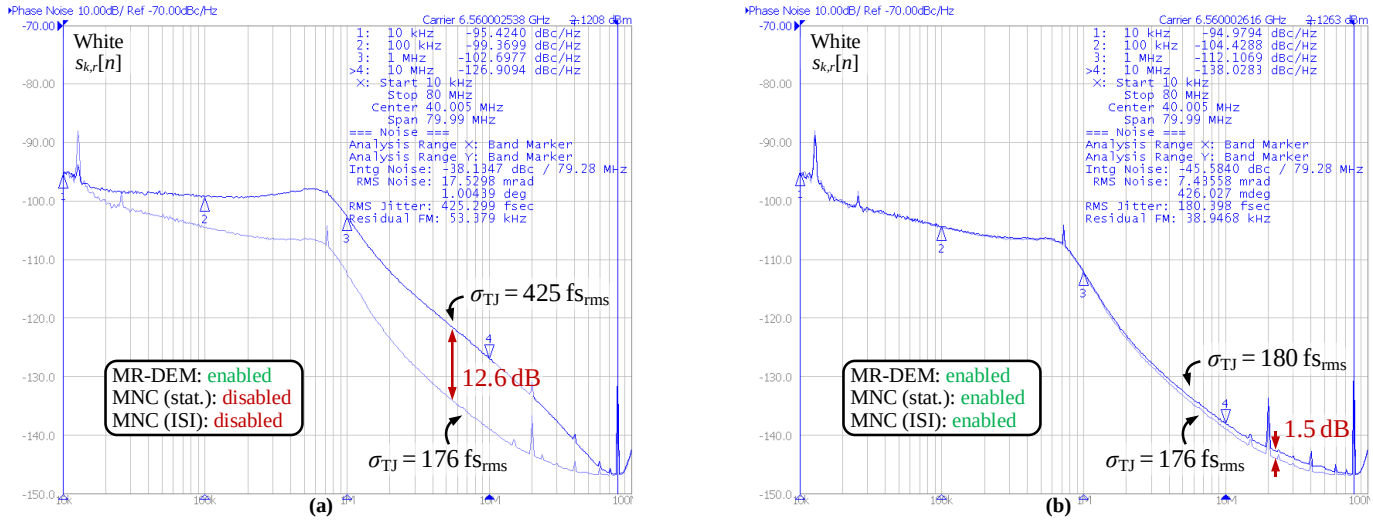

Fig. 11. Measured PLL phase noise at f_{PLL} = 6.56 GHz with MR-DEM enabled for (a) MNC disabled, and (b) MNC (both stat. and ISI branches) enabled.

TABLE I
PNR DIGITAL BLOCK POWER CONSUMPTION

MR-DEM	MNC		Power increase (mW) [1]	
	Stat.-error branch	ISI-error branch	MNC coeffs. update active	MNC coeffs. frozen
Disabled	Enabled	Enabled	-	0.25
First-Order Highpass-Shaped $s_{k,r}[n]$				
Enabled	Disabled	Disabled	0.52	0.52
	Enabled	Disabled	0.92	0.57
	Enabled	Enabled	1.64	0.86
White $s_{k,r}[n]$				
Enabled	Disabled	Disabled	0.43	0.43
	Enabled	Disabled	0.83	0.47
	Enabled	Enabled	1.40	0.70

[1] With respect to the case where MR-DEM and MNC are disabled.

that of the PLL without the MR-DEM and MNC techniques enabled but with considerably smaller FCE mismatches.

The phase noise profile shown in Fig. 9(d) was obtained by setting the MNC loop gains, K_a and K_b in Fig. 4(b), to 2^{-7} and 2^{-8}, respectively. Unfortunately, these are the minimum values of K_a and K_b supported by this design, although it would have been simple to make the design support smaller values. Had smaller values of K_a and K_b been used, it is expected that less "residual" spectral breathing would have occurred.

Fig. 10 shows the measured PLL phase noise at f_{PLL} = 6.56 GHz with the signal source analyzer's averaging option set to 32 and different combinations of the proposed techniques enabled and disabled. The PLL bandwidth was set to 900 kHz in each case. Fig. 10(a) shows the effect on the PLL phase noise of the MR-DEM technique in the absence of MNC compared to the baseline case of Fig. 9(a). As shown in the figure, the spot phase noise increases by up to 11.5 dB compared to the case of Fig. 9(a), whereas σ_{TJ} from 10 kHz to 80 MHz increases from 176 fs$_{rms}$ to 190 fs$_{rms}$. Fig. 10(b) shows the measured PLL phase noise with the MR-DEM technique and only the static-error correction branch of the MNC technique enabled. The spot phase noise around a 20 MHz offset frequency decreases by 8 dB, which corresponds to most of the noise introduced by the MR-DEM technique. Fig. 10(c) shows the measured PLL phase noise with the MR-DEM technique and both branches of the MNC technique enabled. As shown in Fig. 10(c), the improvement after applying ISI error correction results in a 1.2 dB spot phase noise reduction around a 20 MHz offset frequency, and the resulting phase noise profile still shows some residual error, although its effect on σ_{TJ} is almost negligible.

As explained in [21], each $b_{k,r}[n]$ coefficient in (2) converges to a value proportional to the average over $1/f_{fast}$ of its respective $\gamma_{k,r}(t)$ coefficient in (1). Consequently, the ISI-error correction branch of the MNC technique does not perfectly cancel the third term from the right side of (1). Nonetheless, as demonstrated in Fig. 10, this is not a problem in practice because the power of the ISI-error component introduced by the MR-DEM technique is much smaller than that of the static-error component. Additional measurements taken by the authors for the same conditions used to generate Fig. 9(d), except with the MNC technique's ISI-error correction branch disabled, yielded phase noise profiles virtually identical to that shown in Fig. 9(d). This suggests that, at least in the case of the presented prototype IC, the ISI-error correction branch of the MNC technique could be omitted to save both power and area without significantly affecting the effectiveness of the MR-DEM and MNC techniques to mitigate spectral breathing.

Although originally intended to work with highpass-shaped switching sequences, the MNC technique also works with white switching sequences. Fig. 11 shows measured PLL phase noise profiles similar to those shown in Fig. 10(a) and Fig. 10(c) but for white switching sequences within the slow DEM encoder. As shown in Fig. 11(a), the PLL phase noise is severely degraded when enabling MR-DEM in this case. The spot phase noise increases by up to 12.6 dB, and σ_{TJ} increases from 176 to 425 fs$_{rms}$. Nonetheless, as shown in Fig 11(b), the MNC technique cancels most of the noise introduced by the MR-DEM technique, except for a small portion for the reasons explained above. As in the case with highpass-shaped switching sequences, most of the noise in Fig. 11(a) gets cancelled when enabling the MNC technique's static-error correction branch; the ISI-error correction branch accounts for less than a 1 dB reduction.

Table I summarizes the increase of the PNR digital block's power consumption for different combinations of the MR-DEM and MNC techniques enabled and disabled. As shown in the table, the proposed techniques increase the power con-

sumption by up to 1.64 mW and 1.4 mW when using highpass-shaped and white switching sequences, respectively. In both cases, the most significant contributor is the ISI-error correction branch of the MNC technique.

The proposed techniques can be used without significantly increasing the PLL's power consumption by freezing the MNC coefficients after convergence and disabling the MR-DEM technique, in which case the PLL's power consumption only increases by 250 μW. However, as demonstrated in Fig. 9(c) and Fig. 9(d), the effectiveness of the techniques to mitigate spectral breathing is slightly reduced is in this case.

Additional measurements taken by the authors suggest that the proposed techniques have no effect on the PLL's fractional spur performance. In contrast, the reference spur power increases by 10 dB (from –80 dBc to –70 dBc) as a result of enabling MR-DEM, but it does not increase when MNC is enabled with its coefficients frozen and MR-DEM is disabled.

Although the reference spur is expected to increase when enabling MR-DEM due to coupling from the DCO control lines to the DCO outputs, the authors believe that the reported increase of 10 dB is exacerbated by a layout issue in the DCO, which was not caught prior to fabrication because of a post-layout extraction tool flaw. As explained in [4], this issue caused the DCO's quality factor to be significantly lower than expected even after FIB surgery. Consequently, the DCO outputs swing is extremely low, even when raising the DCO tank's supply voltage to 0.9 V, and the DCO is highly sensitive to interference from other circuit blocks.

V. Conclusion

This paper presents the first experimental demonstration of the MR-DEM and MNC techniques described in [21], which mitigate the spectral breathing problem in digital PLLs that results from non-ideal FCE behavior. Additionally, implementation details as well as practical observations that complement the techniques' descriptions in [21] are presented.

The MR-DEM technique linearizes the DCO input-output characteristics at the expense of additive highpass-shaped error which depends on known digital sequences. This error has a component that arises from FCE static gain errors, and another component that arises from ISI, both of which are cancelled by the MNC technique. By freezing the MNC coefficients after convergence and disabling MR-DEM, the presented techniques significantly mitigate the spectral breathing problem while only negligibly increasing the PLL's total power consumption.

Measurement results suggest that for this particular DCO design, the power of the ISI component of the DCO error is much less significant than that of the component that arises from FCE static gain errors. Furthermore, the results suggest that white switching sequences can be used in the MR-DEM technique's slow DEM encoder to reduce digital logic complexity, thereby reducing both power and area consumption.

References

[1] 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.

[2] 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.

[3] C. Weltin-Wu, G. Zhao, I. Galton, "A 3.5 GHz Digital Fractional-*N* Frequency Synthesizer Based on Ring Oscillator Frequency-to-Digital Conversion," *IEEE J. Solid-State Circuits*, vol. 50, no. 12, pp. 2988–3002, Dec. 2015.

[4] E. Helal, E. Alvarez-Fontecilla, A. I. Eissa, and I. Galton, "A Time Amplifier Assisted Frequency-to-Digital Converter Based Digital Fractional-*N* PLL," *IEEE J. Solid-State Circuits*, early access, Feb. 2, 2021, doi: 10.1109/JSSC.2020.3048650.

[5] H. Liu, D. Tang, Z. Sun, W. Deng, H. C. Ngo and K. Okada, "A Sub-mW Fractional- *N* ADPLL With FOM of −246 dB for IoT Applications," *IEEE J. Solid-State Circuits*, vol. 53, no. 12, pp. 3540–3552, Dec. 2018.

[6] C. Li, M. Yuan, C. Liao, Y. Lin, C. Chang and R. B. Staszewski, "All-Digital PLL for Bluetooth Low Energy Using 32.768-kHz Reference Clock and ≤0.45-V Supply," *IEEE J. Solid-State Circuits*, vol. 53, no. 12, pp. 3660–3671, Dec. 2018.

[7] A. Elkholy, S. Saxena, G. Shu, A. Elshazly and P. K. Hanumolu, "Low-Jitter Multi-Output All-Digital Clock Generator Using DTC-Based Open Loop Fractional Dividers," *IEEE J. Solid-State Circuits*, vol. 53, no. 6, pp. 1806–1817, June 2018.

[8] C. Yao *et al.*, "A 14-nm 0.14-psrms Fractional-N Digital PLL With a 0.2-ps Resolution ADC-Assisted Coarse/Fine-Conversion Chopping TDC and TDC Nonlinearity Calibration," *IEEE J. Solid-State Circuits*, vol. 52, no. 12, pp. 3446–3457, Dec. 2017.

[9] M. Talegaonkar *et al.*, "A 5GHz Digital Fractional-*N* PLL Using a 1-bit Delta–Sigma Frequency-to-Digital Converter in 65 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 52, no. 9, pp. 2306–2320, Sept. 2017.

[10] Y. Wu, M. Shahmohammadi, Y. Chen, P. Lu and R. B. Staszewski, "A 3.5–6.8-GHz Wide-Bandwidth DTC-Assisted Fractional-*N* All-Digital PLL With a MASH ΔΣ-TDC for Low In-Band Phase Noise," *IEEE J. Solid-State Circuits*, vol. 52, no. 7, pp. 1885–1903, July 2017.

[11] D. Liao, H. Wang, F. F. Dai, Y. Xu, R. Berenguer and S. M. Hermoso, "An 802.11a/b/g/n Digital Fractional-*N* PLL With Automatic TDC Linearity Calibration for Spur Cancellation," *IEEE J. Solid-State Circuits*, vol. 52, no. 5, pp. 1210–1220, May 2017.

[12] D. Pfaff, R. Abbott, X. Wang, S. Moazzeni, R. Mason and R. R. Smith, "A 14-GHz Bang-Bang Digital PLL With Sub-150-fs Integrated Jitter for Wireline Applications in 7-nm FinFET CMOS," *IEEE J. Solid-State Circuits*, vol. 55, no. 3, pp. 580–591, March 2020.

[13] A. Santiccioli, M. Mercandelli, A. L. Lacaita, C. Samori and S. Levantino, "A 1.6-to-3.0-GHz Fractional-*N* MDLL With a Digital-to-Time Converter Range-Reduction Technique Achieving 397-fs Jitter at 2.5-mW Power," *IEEE J. Solid-State Circuits*, vol. 54, no. 11, pp. 3149–3160, Nov. 2019.

[14] H. Liu *et al.*, "A 265-μW Fractional-*N* Digital PLL With Seamless Automatic Switching Sub-Sampling/Sampling Feedback Path and Duty-Cycled Frequency-Locked Loop in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 54, no. 12, pp. 3478–3492, Dec. 2019.

[15] L. Bertulessi, L. Grimaldi, D. Cherniak, C. Samori and S. Levantino, "A Low-Phase-Noise Digital Bang-Bang PLL with Fast Lock Over a Wide Lock Range," *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, pp. 252–254. Feb. 2018.

[16] S. Kim, S. Hong, K. Chang, H. Ju, J. Shin, B. Kim, H.-J. Park, and J.-Y. Sim, "A 2 GHz Synthesized Fractional-*N* ADPLL With Dual-Referenced Interpolating TDC," *J. Solid-State Circuits*, vol. 51, no. 2, pp. 391–400 Feb. 2016.

[17] M. Dayanik and M. Flynn, "Digital Fractional-*N* PLLs Based on a Continuous-Time Third-Order Noise-Shaping Time-to-Digital Converter for a 240-GHz FMCW Radar System," *J. Solid-State Circuits*, vol. 53, no. 6, pp.1719–1730 , June 2018.

[18] A. Elkholy, T. Anand, W.-S. Choi, 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," *J. Solid-State Circuits*, vol. 50, no. 4, pp. 867–881, April 2015.

[19] D. Weyer, M. B. Dayanik, S. Jang, and M. P. Flynn, "A 36.3-to-38.2GHz −216dBc/Hz2 40nm CMOS Fractional-*N* FMCW Chirp Synthesizer PLL with a Continuous-Time Bandpass Delta-Sigma Time-to-Digital Con-

verter," *IEEE Int. Solid-State Circuits Cong. (ISSCC) Dig. Tech. Papers*, pp. 250–252, Feb. 2018.
[20] A. Santiccioli *et al.*, "A 66-fs-rms Jitter 12.8-to-15.2-GHz Fractional-*N* Bang–Bang PLL With Digital Frequency-Error Recovery for Fast Locking," *IEEE J. Solid-State Circuits*, vol. 55, no. 12, pp. 3349–3361, Dec. 2020.
[21] E. Alvarez-Fontecilla, C. Venerus and I. Galton, "Multi-Rate DEM With Mismatch-Noise Cancellation for DCOs in Digital PLLs," *IEEE Trans. Circuits and Syst. I. Reg. Papers*, vol. 65, no. 10, pp. 3125–3137, Oct. 2018.
[22] O. Eliezer, B. Staszewski, J Mehta, F. Jabbar, I. Bashir, "Accurate Self-Characterization of Mismatches in a Capacitor Array of a Digitally-Controlled Oscillator," *IEEE Dallas Circuits and Syst. Workshop*, pp. 17-18, Oct. 2010.
[23] O. Eliezer, B. Staszewski, S Vemulapalli, "Digitally controlled oscillator in a 65nm GSM/EDGE transceiver with built-in compensation for capacitor mismatches," *IEEE Radio Frequency Integrated Circuits Symp.*, pp. 5-7 June 2011.
[24] I. Galton, "Why Dynamic-Element-Matching DACs Work," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 2, pp. 69-74, Mar. 2010.
[25] E. Alvarez-Fontecilla, A. I. Eissa, E. Helal C. Weltin-Wu and I. Galton, "Delta-Sigma FDC Enhancements for FDC-Based Digital Fractional-*N* PLLs," *IEEE Trans. Circuits Syst. I. Reg. Papers*, vol. 68, no. 3, pp. 965–974, March 2021.
[26] P. Kiss, J. Arias, D. Li and V. Boccuzzi, "Stable High-Order Delta-Sigma Digital-to-Analog Converters," *IEEE Trans. Circuits and Syst. I. Reg. Papers*, vol. 51, no. 1, pp. 200–205, Jan. 2004
[27] I. Galton, "Spectral shaping of circuit errors in digital-to-analog converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 10, pp. 808–817, Oct. 1997.
[28] J. Welz, I. Galton, and E. Fogleman, "Simplified logic for first-order and second-order mismatch-shaping digital-to-analog converters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 48, no. 11, pp. 1014–1028, Nov. 2001.
[29] C. Venerus, J. Remple, and I. Galton, "Simplified Logic for Tree-Structure Segmented DEM Encoders," *IEEE Trans. Circuits and Syst. II. Exp. Briefs*, vol. 63, no. 11, pp. 1029–1033, Nov. 2016.
[30] 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.
[31] B. Razavi, *RF Microelectronics*, Upper Saddle River, NJ: Prentice Hall, 1998.

Enrique Alvarez-Fontecilla received the B.Sc. and M.Sc. degrees from Pontificia Universidad Católica de Chile (PUC), Santiago, Chile, in 2011 and 2013, respectively, and the Ph.D. degree from the University of California at San Diego, San Diego, CA, USA in 2021, all in electrical engineering. From 2012 to 2015, he was an Adjunct Assistant Professor at the School of Engineering and the Institute of Philosophy, PUC. He is currently an Algorithm Design Engineer with Analog Devices Inc., San Diego, CA, USA. He is also de Founder of Castor Barracuda.

Eslam Helal received the B.Sc. (Hons) and M.Sc. degrees in electrical engineering from Ain Shams University (ASU), Cairo, Egypt, in 2014 and 2018, respectively. From 2014 to 2018 he was a part-time analog design engineer at Si-Ware Systems (now Goodix Technology), Cairo, Egypt. Also, he was a teaching and research assistant with the Electronics and Communications Engineering Department at ASU. He is currently pursuing the Ph.D. degree with the University of California at San Diego, San Diego, CA. His research interests include analog/mixed-signal integrated circuits and systems, frequency synthesizers and data converters.

Amr I. Eissa received the B.Sc. (Hons.) and M.Sc. degrees in electrical engineering from Ain Shams University (ASU), Cairo, Egypt, in 2012 and 2016, respectively. Currently, he is pursuing the Ph.D. degree in electronic circuits and systems at the University of California, San Diego. From 2013 to 2016, he was a Teaching and Research Assistant with the Electronics and Communications Engineering Department, ASU. His research interests include the analysis and design of analog/mixed-signal integrated circuits and systems.

Ian Galton received the B.Sc. degree from Brown University in 1984, and the M.S. and Ph.D. degrees from the California Institute of Technology in 1989 and 1992, respectively, all in electrical engineering. Since 1996, he has been a Professor of electrical engineering with the University of California, San Diego, where he teaches and conducts research in the field of mixed-signal integrated circuits and systems for communications. His research involves the invention, analysis, and integrated circuit implementation of critical communication system blocks such as data converters and phase-locked loops.