A Reconfigurable DLL-Based Digital-to-Time Converter Using Charge Pump Current Interpolation and Digital Predistortion Linearization

Ahmed F. Elian, *Student Member, IEEE*, Ibrahim M. Elfadel, *Senior Member, IEEE*, and Ayman Shabra, *Member, IEEE*

***Abstract*—This brief presents a digital-to-time converter (DTC) based on a reconfigurable delay-locked loop that employs dual feedback taps and interpolation at the output of two charge pumps biased with programmable complementary currents from a current digital-to-analog converter (I-DAC) to achieve fine delay tuning. Through selection of the feedback and output delay line taps, the 65-nm CMOS prototype can be configured to achieve different specs, such as a 24.5 ps delay span with a 0.46 ps maximum delay step and 1.33 ps maximum RMS jitter or a span of 244 ps with a 3.30 ps maximum step and 2.63 ps maximum RMS jitter when a 7-bit I-DAC and a 2 GHz input are used. A simple digital predistortion method to compensate for the inherent nonlinearity of the architecture is presented and experimentally shown to improve the worst-case integral nonlinearity from −14.3 LSB (−27.6 ps) to +1.79 LSB (+3.43 ps) for the 244 ps delay span case, and by 67%–88% at all taps for the same loop configuration. Excluding the I-DAC, which would require an estimated 0.2 mW, the DTC consumes 0.665 mW from a 1.25-V supply at 2 GHz and occupies an area of 46 μm × 28 μm.**

***Index Terms*—Delay-locked loop, DTC, charge pump, embedded phase interpolation, predistortion, tunable delay, voltage-controlled delay line.**

I. Introduction

DIGITAL-TO-TIME conversion (DTC) typically requires a coarse and a fine adjustment. Coarse delay steps can be realized by dividing a high-frequency clock [1], [2], by using a delay-locked loop (DLL) [3] or by using a multiphase oscillator [4]. To achieve a delay step finer than the minimum delay of a buffer stage, it is possible to employ techniques such as analog phase interpolators (PI) [1], [2], delay lines (DL) with either a programmable capacitive loading [5] or an adjustable signal swing and a constant edge slope [6]. The statistical variation of the delay of integrated clock buffers in advanced nodes can also be exploited to achieve fine delay tuning [7].

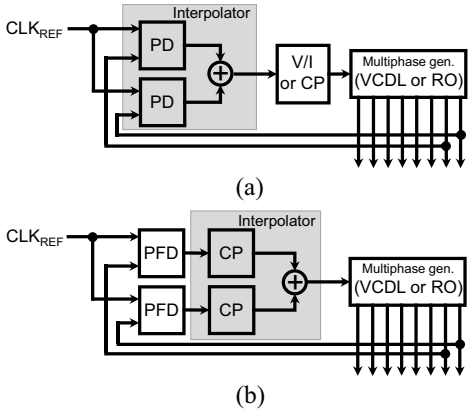

Fig. 1. Delay tuning in a DLL or RO PLL by interpolation at the outputs of (a) phase detectors, or (b) charge pumps.

Improved robustness of delay against process, voltage and temperature (PVT) variations is offered by a closed-loop system such as a DLL or a ring oscillator (RO) phase-locked loop (PLL). The DLL and RO PLL reported in [8] and [9], respectively, demonstrated a fine delay step. This is achieved by using two clock phases from two voltage-controlled delay line (VCDL) or RO taps as feedback signals to drive different phase detectors (PD), as shown in Fig. 1a. A programmable interpolator then generates a control voltage for the VCDL or RO such that in steady state, a virtual, adjustable edge that lies between the two selected feedback phases is locked to the reference. To enable interpolation, [8] and [9] use current-mode logic (CML)-based XOR phase detectors with a programmable bias current, but this results in static currents and increases power consumption. This can be mitigated using the charge pump (CP) interpolation method in Fig. 1b, which was first reported for a RO PLL in [10] and [11].

In this brief, we apply charge pump interpolation to a DLL, which allows us to achieve a reduction in power consumption compared to prior designs in a very compact area and with a comparable delay resolution. In contrast to implementing CP interpolation in a RO PLL [10], [11], where the RO's noise accumulates in it by circulation and is only high-pass filtered by the loop, using a DLL means using a delay line that does

Manuscript received February 24, 2019; accepted March 25, 2019. Date of publication April 4, 2019; date of current version April 30, 2019. This work was supported by the Semiconductor Research Corporation through the MEES I Program under Contract 2011-HJ-2186, and through the ACE4S Program under Contract 2013-HJ-2440. This brief was recommended by Associate Editor J. M. de la Rosa. *(Corresponding author: Ibrahim M. Elfadel.)*

A. F. Elian was with the Institute Center for Microsystems, Masdar Institute, Abu Dhabi 54224, UAE. He is now with the University of Toronto, Toronto, ON M5S 1A1, Canada (e-mail: ahmed.elian@mail.utoronto.ca).

I. M. Elfadel is with the Department of Electrical and Computer Engineering, Khalifa University, Abu Dhabi 54224, UAE (e-mail: ielfadel@masdar.ac.ae).

A. Shabra was with the Institute Center for Microsystems, Masdar Institute, Abu Dhabi 54224, UAE. He is now with MediaTek, Woburn, MA 01801 USA (e-mail: ayman.shabra@mediatek.com).

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/TCSII.2019.2909431

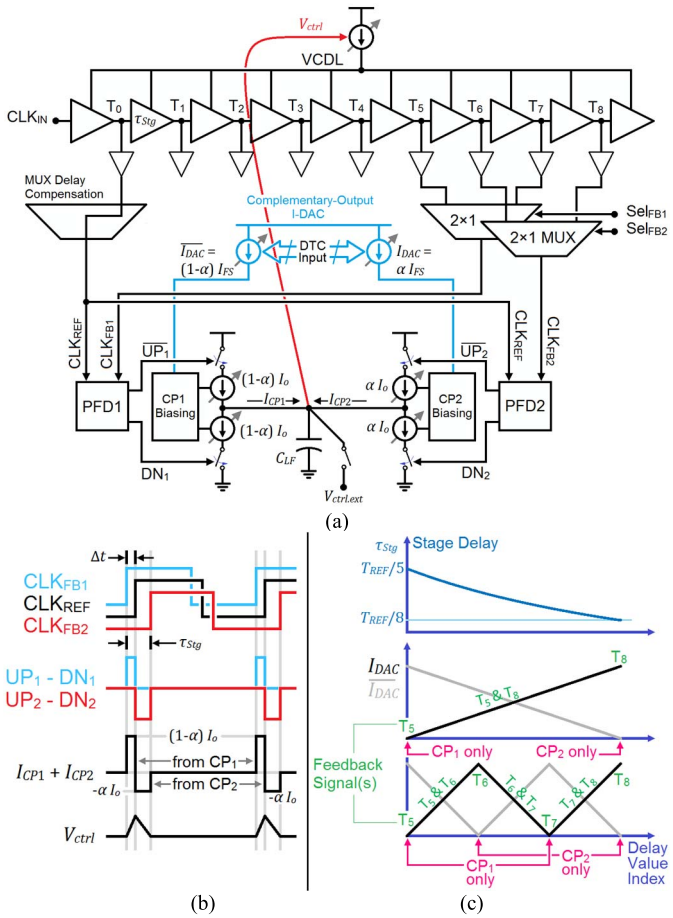

Fig. 2. (a) Block diagram of the proposed DTC, (b) ideal steady-state waveforms illustrating the interpolation technique and (c) an illustration of how delay sweeping is done from locking T_5 to locking T_8 by selecting feedback signals and adjusting the I-DAC.

not circulate noise. This is beneficial in reducing jitter and/or power, especially when the required delay span is small.

We address the inherent, system-level delay nonlinearity of the interpolating DLL [8] using a simple digital predistortion method that can be implemented in a look-up table, and we experimentally demonstrate its effectiveness at improving the integral and differential nonlinearities (INL and DNL).

To achieve an arbitrarily large range, the DTC we present could be modified to use a sub-ranging technique such as the one presented in [12], where the two-step design is comprised of a counter-based coarse stage and a calibrated delay line fine stage setting the final resolution.

This brief is organized as follows. Section II presents the architecture and the interpolation method. In Section III, we derive a linearity enhancement method that is straightforward to implement. In Section IV, we present our 65 nm silicon results, including delay, jitter and nonlinearity measurements. We also give a thorough silicon-based performance summary of our reconfigurable DTC design along with extensive comparisons with the state of the art.

II. Architecture

The block diagram of the proposed DTC is shown in Fig. 2a. It is composed of a voltage-controlled delay line (VCDL), two phase/frequency detectors (PFD) and two charge pumps (CP). The CPs are biased by a complementary-output current digital-to-analog converter (I-DAC) resulting in CP output stage bias currents of αI_o and $(1-\alpha)I_o$. The feedback signals, $\mathrm{CLK_{FB1}}$ and $\mathrm{CLK_{FB2}}$, that connect the VCDL to the PFD inputs, are selected by two 2×1 loop multiplexers (MUX), resulting in four possible loop configurations. The PFDs compare these feedback signals to the input reference clock and a weighted sum of the errors is integrated on the loop capacitor C_{LF} by the CPs to form the VCDL's control voltage, V_{ctrl}.

As illustrated in Fig. 2b, the delays between $\mathrm{CLK_{FB1}}$, CLK_{REF} and $\mathrm{CLK_{FB2}}$ in steady state are defined by programming the CP currents to αI_o and $(1-\alpha)I_o$ and by charge conservation at node V_{ctrl}. If $\mathrm{CLK_{FB1}}$ and $\mathrm{CLK_{FB2}}$ are selected from taps N_{L0} and $N_{L0}+N_B$, we can use charge conservation to write the delay of a single VCDL stage, τ_{Stg}, under lock condition:

$$\begin{aligned}(1-\alpha)I_o\Delta t &= \alpha I_o\big(N_B\tau_{Stg}-\Delta t\big) \rightarrow \Delta t = \alpha N_B\tau_{Stg}\\ T_{REF} &= N_{L0}\tau_{Stg}+\Delta t = (N_{L0}+\alpha N_B)\tau_{Stg}\\ \tau_{Stg} &= \frac{T_{REF}}{N_{L0}+\alpha N_B} = \frac{T_{REF}}{N_L}\end{aligned} \tag{1}$$

where T_{REF} is the input clock period and N_L is the effective number of delay stages within the loop. Thanks to interpolation, N_L can be virtually fractional with integer and fractional parts. N_B is the number of delay stages between $\mathrm{CLK_{FB1}}$ and $\mathrm{CLK_{FB2}}$ and is an integer greater than or equal to 1.

For our specific case shown in Fig. 2a, the four possible loop configurations correspond to (N_{L0}, N_B) values of (5, 1), (6, 1), (7, 1) and (5, 3). Fig. 2c shows how delay can be swept by reconfiguring the loop and adjusting the I-DAC. In the top plot in Fig. 2c, it is shown how interpolation can sweep τ_{Stg} between $T_{REF}/5$ and $T_{REF}/8$, in two ways; the first is by selecting tap 5 (T_5) and tap 8 (T_8) as feedback taps and the second is by selecting the feedback taps sequentially as (T_5, T_6), then (T_6, T_7) and finally (T_7, T_8). The middle and bottom Fig. 2c plots show how the complementary I-DAC currents in these two scenarios, respectively, are adjusted to achieve this delay sweeping. To minimize delay discontinuities at points where the feedback taps are changed to/from (T_6, T_7), we only change the feedback clock that drives the CP biased at zero current. As a result, $\mathrm{CLK_{FB1}}$ leads $\mathrm{CLK_{FB2}}$ in all but the (T_6, T_7) configuration.

Unlike the XOR-based PD in [8] and [9], we use a simple dynamic tri-state PFD, as in [13], to save power. Our CP is similar to [14] with enhanced output impedance to reduce up/down current imbalance. This source-switched CP topology helps isolate the CP output node (V_{ctrl}) from its input pulses, reducing VCDL disturbance.

III. Linearity Enhancement by Predistortion

Digital predistortion has been used in literature to correct for the nonlinearity of DTCs. In [15], for example, it is used to linearize the characteristic of a DTC used within a digital PLL. The predistortion coefficients are adaptively generated and a piece-wise linear approximation is used to reduce hardware complexity. In our DTC, the delay of a single stage (τ_{Stg}), as can be seen from (1), is a nonlinear function of α and this becomes more pronounced for larger values of N_B. Since α is the ratio of the I-DAC's digital input to the digital full-scale, this relationship relates the digital input and the output delay.

Since (1) represents a memoryless nonlinearity, it is straightforward to show that this nonlinearity can be compensated

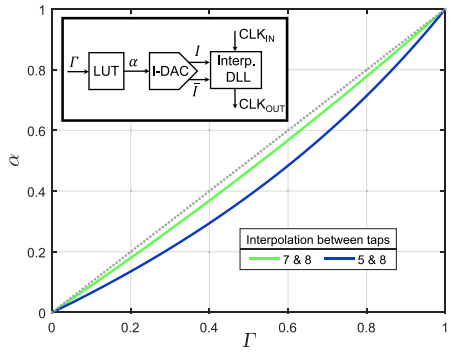

Fig. 3. The relationship between α and Γ for two feedback configurations. Inset shows a block diagram including the LUT for linearity enhancement.

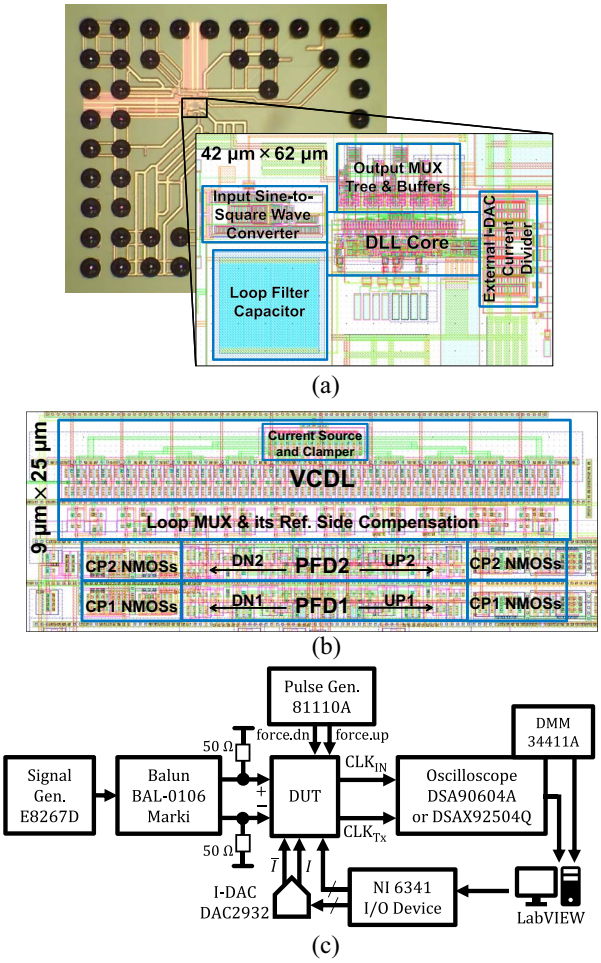

Fig. 4. (a) Die photo and layout of the circuit, (b) layout details of the DLL core and (c) a simplified block diagram of the experimental setup.

exactly by the following variable transformation:

$$\alpha = \frac{N_{L0}\Gamma}{N_{L0} + N_B(1-\Gamma)} \tag{2}$$

Substituting this for α in (1), we obtain a linear relationship between the delay τ_{Stg} and Γ:

$$\tau_{Stg} = \frac{T_{REF}}{N_{L0}}\left(1 - \frac{N_B}{N_{L0}+N_B}\Gamma\right) \tag{3}$$

Like α, Γ ranges from 0 to 1 and is the ratio of a digital code to the digital full scale, making (2) a unique solution. Thus, linearity can be improved by using Γ as the digital input to be adjusted in uniform steps, transforming it to α using (2) and using α (the predistorted codes) to adjust the I-DAC currents. The nonlinear Γ-to-α transformation, plotted in Fig. 3, can be implemented in a digital look-up table (LUT) which simply maps digital codes based on (2), as shown in the inset of Fig. 3. The hardware implementation of the LUT can be as simple as a read-only memory (ROM) with low power consumption and small area. To reduce the ROM size, only the difference between α and Γ can be stored.

It is worth noting that this correction method is insensitive to offset and gain mismatches between the model and the actual delay. The substitution in (2) will still give a linear $\tau_{Stg}(\Gamma)$ with any actual τ_{Stg} that follows (2) even with an arbitrary offset and/or multiplication factor, and the LUT codes derived from it will therefore linearize the delay effectively. However, since this method is not dependent on calibration, it will not account for residual nonlinearity errors due to analog circuit non-idealities such as CP up/down mismatch and limited speed. To reduce the residual nonlinearity, it is best to reduce these circuit nonidealities by improving the circuit implementation while still using the exact same digital predistortion since the latter will be effective in correcting the system-level nonlinearity independently from the quality of the circuit implementation. As we demonstrate in the experimental results, the reduction in the overall nonlinearity due to (2) alone is significant.

IV. Experimental Results

The prototype was manufactured in a 65 nm bulk CMOS process and the die photo is shown in Fig. 4a. The rectangular area enclosing the DLL core, the details of which are shown in Fig. 4b, and the loop filter capacitor (0.45 pF) occupies 46 μm × 28 μm only. Since the I-DAC is low-risk, the CP biasing currents were supplied from an off-chip 12-bit accurate I-DAC IC to produce 129 or 513 levels and a full-scale current of 128 μA with INL $<$ 45 nA. A simplified diagram of the experimental setup is shown in Fig. 4c.

All measurement results reported in this section are at 2 GHz and, unless otherwise stated, all jitter measurements reported are the contribution of the DLL only, estimated by subtracting the mean-square jitter measured for CLK_{IN}. Delay measurements used averaging over 20,000 or 30,000 acquisitions, and measurements of RMS of absolute jitter were collected over 20,000 hits.

Fig. 5 shows delay and jitter measurement results without linearization for taps 1 and 2 over two full delay sweeps using a 129-level I-DAC. In Fig. 5a, interpolation is between the consecutive taps (T_7, T_8), (T_6, T_7) and (T_5, T_6), while in Fig. 5b, it is between taps 5 and 8. Fig. 6 shows the measured worst-case jitter for tap 1 when interpolating between taps 5 and 6 and corresponds to point $\oplus$ in Fig. 5. Note that the discontinuities in the delay characteristics seen in Fig. 5a are eliminated in the delay plot in Fig. 5b because the feedback tap does not need to be changed mid-sweep.

However, since in Fig. 5b interpolation is between taps 5 and 8, while the widest delay span is achieved, the measured delay exhibits a pronounced nonlinearity. This is visible for taps 4 and 7 in the blue (dark) INL vs. code curves in Fig. 7a, where a 7-bit I-DAC is used. INL is visibly dominated by the characteristic in (1). Fig. 7a also demonstrates in red (light) the INL and DNL achieved at the same taps using the predistortion transformation in (2) starting with a 9-bit I-DAC,

TABLE I
PERFORMANCE SUMMARY AND COMPARISON

		This Work					[1] 2016	[2] 2013		[5] 2014	[8] 2011		[12] 2018
Process		65 nm CMOS					28 nm CMOS	65 nm CMOS		28 nm CMOS	0.13 µm SiGe BiCMOS		65 nm CMOS
VDD (V)		1.25					1.1	1.2		0.9			1.05
Input Frequency (GHz)		2					8	0.4-6		0.04	1		3.2
								2	6				
Output Frequency (GHz)		2					2	0.1-1.5		0.04	1		0.1
								0.5	1.5				
No. of Levels		128					2048	256		1024	64	32	16384
Output Tap		2	1	4	4	6	N/A	N/A		N/A	10		N/A
Feedback Taps		6, 7	5, 6	5, 6	5, 8	5, 8	N/A	N/A		N/A	(14,15) (15,16)	14, 16	N/A
Raw / Linearized by Predistortion		Raw	Raw	Raw	Lin.	Lin.					Raw		
Delay Step (ps)	Mean (LSB size)	0.19	0.16	0.58	1.33	1.76	0.244	7.8 [b]	2.6 [b]	0.55	1.36	1.85	0.33
	Max	0.46	0.43	1.32	2.32	2.58	0.55 [d]	12 [b]	16 [b]	0.86 [b]	5	4	1.33 [b]
Delay Span (ps)		24.5	19.8	73.0	169	224	500	2000	667	550	50		5310
Max/Min INL or Max |INL|	(LSB)	2.14/ -1.86	7.11/ -1.11	0.23/ -4.67	0.30/ -2.28	3.35/ -0.21	1.4/-4.9 [b]	±1.33	7.1 [b]	1.8/-0.7			3.8/-5.0 [b]
	(ps)	0.41/ -0.36	1.11/ -0.17	0.13/ -2.68	0.40/ -3.03	5.90/ -0.38	0.33/-1.2	±10.4 [b]	19 [b]	1.0/-0.4 [b]			1.25/-1.65
Max/Min DNL or Max |DNL|	(LSB)	1.40/ -1.17	1.78/ -2.01	1.29/ -0.73	0.74/ -0.55	0.46/ -0.66	1.25/-0.26	0.52/ -0.2	5 [b]	0.6/-0.7	2.7/ ( ) [b]	1.2/ ( ) [b]	3.0/-1.4 [b]
	(ps)	0.27/ -0.23	0.28/ -0.31	0.74/ -0.42	0.99/ -0.74	0.81/ -1.16	0.305/ -0.063	4.1/ -1.6 [b]	13 [b]	0.3/-0.4 [b]	3.6/ ( ) [b]	2.1/ ( ) [b]	1.00/-0.45
RMS Jitter (ps)	RMS over Pts.	1.25	1.19	1.41	1.25	1.67		3.37 [c]			0.86	0.82	0.34
	Max	1.33	1.33	1.63	1.62	2.42		3.37 [c]			1.3	1	
Power (mW)		0.637-0.665 [a]					19.8		4.30	0.52	33		10.13
FOM (fJ/conv) [d]		8.4 [d]	28 [d]	18 [d]	9.0 [d]	13 [d]	24 [b]		82 [b]	24 [b]			31
Area (µm^2)		1.29k [a]					9k	60k		40k	185k		84k
Architecture		Interpolating DLL					Freq. Divider+PI	PI + Harmonic Rejection		Capacitively Loaded DL	Interpolating DLL		Counter + DCDL

[a] Do not include the I-DAC and predistortion circuits, which were not included on-chip. The I-DAC power is estimated to be 0.2 mW.
[b] Calculated. [c] Includes the contribution of jitter on the input clocks, which is at 3.11 ps RMS.
[d] As defined in [12]: FOM $= P/f_{out} \times |\text{INL}|_{pk}/\text{Span}$. For this work's FOM, we used P = 1 mW, leaving a margin for the I-DAC and predistortion circuits.

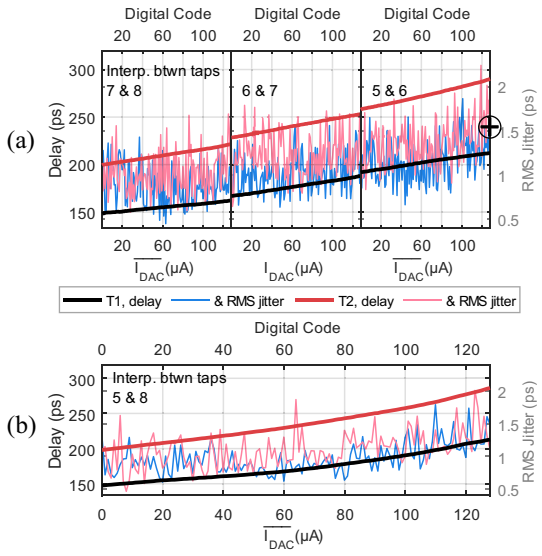

Fig. 5. Delay and jitter measurements vs. I-DAC current for taps 1 and 2 for two full delay sweeps using a 129-level I-DAC without linearization.

Fig. 6. Measured jitter for T_1, including input jitter and output MUX jitter, when interpolating between T_5 and T_6 for the largest delay value marked ⊕ in Fig. 5.

plotted vs. digital code (Γ). The residual nonlinearity after linearization is most likely dominated by the nonlinearity of the charge pumps. Fig. 7b summarizes the worst-case measured INL and DNL in LSB for all DLL taps before and after the linearity enhancement. On the right y-axis of the same plots, the percentage improvement in worst-case INL and DNL after linearization is quantified. The improvement in INL is in the range 67-88% and in DNL is 14-50%. Note that the reported worst-case delay step, DNL and INL values are a bit pessimistic due to being vulnerable to residual jitter observed to be present in delay measurements even after scope averaging.

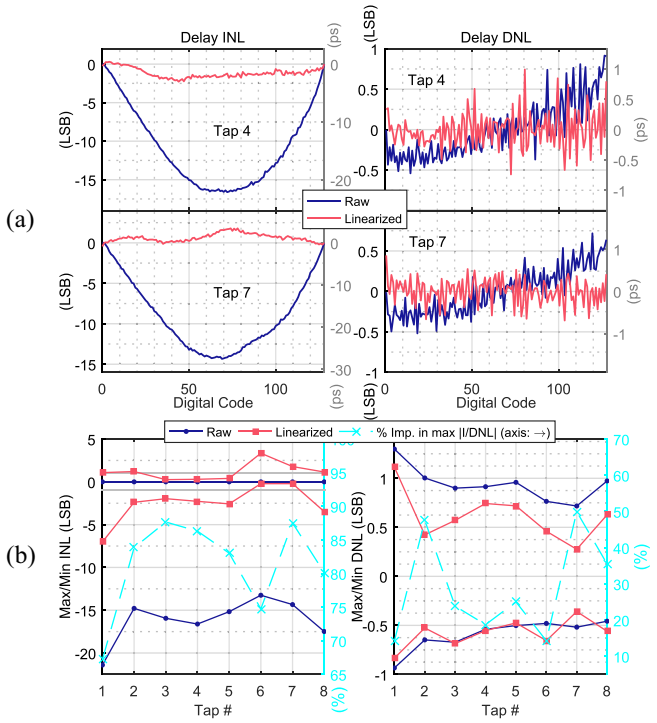

Fig. 7. INL and DNL measurement results with and without linearization when interpolating between taps 5 and 8. (a) shows INL and DNL in LSB and ps vs. code for Taps 4 and 7, and (b) is a summary of maximum and minimum INL and DNL at all taps (left y-axis) and the percent improvement in worst-case INL and DNL through linearization (right y-axis).

The DLL core consumes 0.637-0.665 mW from a 1.25 V supply, not including the power needed for the I-DAC, which is estimated to be around 0.2 mW. This I-DAC power estimate is corroborated by power- and area-efficient programable current circuits reported in literature, such as in [16] and [17]. Of the measured DLL core power, 0.22-0.27 mW is estimated from open-loop VCDL measurements to be consumed by the VCDL. Our measurements clearly show that the delay resolution is jitter-limited. Lowering the jitter level would require increasing the power consumption of the VCDL. Table I summarizes the performance of the DLL, focusing on five configuration-output tap combinations, and provides a detailed comparison with recent literature. The DTC demonstrates a significant reduction in the FOM defined in [12] compared to the state of the art.

V. Conclusion

This brief has reported on the architecture and silicon measurements of a new DTC design applying charge pump current interpolation to a DLL in 65 nm technology. By interpolating between two current-domain phase error signals from two VCDL taps, the 2 GHz DLL finely tunes the delay. The DLL can be configured to achieve several different delay span and delay resolution combinations and can offer sub-ps delay resolution for applications that do not require a large delay span. The presented DTC features a very compact area and a sub-mW power consumption. A simple, calibration-free predistortion method to compensate for the inherent delay nonlinearity of the architecture independently from DLL component non-idealities has also been presented and experimentally shown to improve the worst-case INL by 67-88% for the loop configuration with the widest delay span and the most pronounced nonlinearity.

Acknowledgment

The design, verification and characterization work was conducted at Masdar Institute (now part of Khalifa University), Abu Dhabi, UAE.

References

[1] S. Sievert *et al.*, "2.9. A 2GHz 244fs-resolution 1.2ps-peak-INL edge-interpolator-based digital-to-time converter in 28nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2016, pp. 52–54.

[2] M.-S. Chen, A. A. Hafez, and C.-K. K. Yang, "A 0.1–1.5 GHz 8-bit inverter-based digital-to-phase converter using harmonic rejection," *IEEE J. Solid-State Circuits*, vol. 48, no. 11, pp. 2681–2692, Nov. 2013.

[3] T. Fischer, J. Desai, B. Doyle, S. Naffziger, and B. Patella, "A 90-nm variable frequency clock system for a power-managed Itanium architecture processor," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 218–228, Jan. 2006.

[4] D. Liao, F. F. Dai, B. Nauta, and E. Klumperink, "Multi-phase subsampling fractional-N PLL with soft loop switching for fast robust locking," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Austin, TX, USA, May 2017, pp. 1–4.

[5] N. Markulic, K. Raczkowski, P. Wambacq, and J. Craninckx, "A 10-bit, 550-fs step digital-to-time converter in 28nm CMOS," in *Proc. Eur. Solid State Circuits Conf. (ESSCIRC)*, Sep. 2014, pp. 79–82.

[6] J. Z. Ru, C. Palattella, P. Geraedts, E. Klumperink, and B. Nauta, "A high-linearity digital-to-time converter technique: Constant-slope charging," *IEEE J. Solid-State Circuits*, vol. 50, no. 6, pp. 1412–1423, Jun. 2015.

[7] V. H.-C. Chen and L. Pileggi, "A 69.5 mW 20 GS/s 6b time-interleaved ADC with embedded time-to-digital calibration in 32 nm CMOS SOI," *IEEE J. Solid-State Circuits*, vol. 49, no. 12, pp. 2891–2901, Dec. 2014.

[8] S. Callender and A. M. Niknejad, "A phase-adjustable delay-locked loop utilizing embedded phase interpolation," in *Proc. IEEE Radio Freq. Integr. Circuits Symp. (RFIC)*, Baltimore, MD, USA, Jun. 2011, pp. 1–4.

[9] T. Toifl *et al.*, "A 0.94-ps-RMS-jitter 0.016-mm2 2.5-GHz multiphase generator PLL with 360° digitally programmable phase shift for 10-Gb/s serial links," *IEEE J. Solid-State Circuits*, vol. 40, no. 12, pp. 2700–2712, Dec. 2005.

[10] J.-H. Bae, K.-H. Kim, S. Kim, K.-W. Kwon, and J.-H. Chun, "A low-power dual-PFD phase-rotating PLL with a PFD controller for 5Gb/s serial links," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Seoul, South Korea, May 2012, pp. 2159–2162.

[11] S. Kim, D. Lee, Y.-S. Park, Y. Moon, and D. Shim, "A dual PFD phase rotating multi-phase PLL for 5Gbps PCI express Gen2 multi-lane serial link receiver in 0.13um CMOS," in *Proc. IEEE Symp. VLSI Circuits Tech. Papers*, Kyoto, Japan, Jun. 2007, pp. 234–235.

[12] A. Elmallah, M. G. Ahmed, A. Elkholy, W. Choi, and P. K. Hanumolu, "A 1.6ps peak-INL 5.3ns range two-step digital-to-time converter in 65nm CMOS," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, San Diego, CA, USA, Apr. 2018, pp. 1–4.

[13] F.-R. Liao and S.-S. Lu, "A programmable edge-combining DLL with a current-splitting charge pump for spur suppression," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 12, pp. 946–950, Dec. 2010.

[14] R. C.-H. Chang, H.-M. Chen, and P.-J. Huang, "A multiphase-output delay-locked loop with a novel start-controlled phase/frequency detector," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 9, pp. 2483–2490, Oct. 2008.

[15] 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.

[16] Y. Okuma *et al.*, "0.5-V input digital LDO with 98.7% current efficiency and 2.7-ţA quiescent current in 65nm CMOS," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, San Jose, CA, USA, Sep. 2010, pp. 1–4.

[17] L. G. Salem, J. Warchall, and P. P. Mercier, "20.3 A 100nA-to-2mA successive-approximation digital LDO with PD compensation and sub-LSB duty control achieving a 15.1ns response time at 0.5V," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2017, pp. 340–341.