A 0.52/1 V Fast Lock-in ADPLL for Supporting Dynamic Voltage and Frequency Scaling

Ching-Che Chung, Wei-Siang Su, and Chi-Kuang Lo

***Abstract*— In energy-efficient processing platforms, such as wearable sensors and implantable medical devices, dynamic voltage and frequency scaling allows optimizing the energy efficiency under various modes of operation. The clock generator used in these platforms should be capable of achieving a faster settling time and has a wider operating voltage range. In this brief, a fast lock-in all-digital phase-locked loop (ADPLL) with two operation modes (0.52/1 V) is presented. The proposed ADPLL can quickly compute the desired digitally controlled oscillator control code with high accuracy. Therefore, the proposed ADPLL can achieve a fast setting time with frequency errors <5% within four clock cycles. The proposed ADPLL is implemented using a standard performance 90-nm CMOS process. The output frequency of the ADPLL ranges from 60 to 600 MHz at 1 V, and from 30 to 120 MHz at 0.52 V. The power consumption of the proposed ADPLL is 0.92 mW at (1 V, 600 MHz), and 37 μW at (0.52 V, 120 MHz).**

***Index Terms*— All-digital phase-locked loop (ADPLL), digitally controlled oscillator (DCO), dynamic voltage and frequency scaling (DVFS), fast lock-in, low power.**

I. Introduction

In recent years, energy-efficient processing platforms have been developed for wearable sensors [1], implantable medical devices, and wireless sensors [2]. Dynamic voltage and frequency scaling (DVFS) allows optimizing the energy efficiency under various modes of operation. For quickly recovering from the low-power modes to the normal mode of operation, the clock generator should be capable of achieving faster lock-in.

When the power supply is reduced to ∼0.5 V, the voltage headroom is insufficient for a voltage-controlled oscillator (VCO) to provide an acceptable frequency range with a reasonable VCO gain (K_{VCO}). Therefore, in low-voltage phase-locked loops (PLLs) [4]–[6], multiband VCOs are employed to reduce the K_{VCO} value. However, multiband VCOs [4]–[6] or multiband digitally controlled oscillators (DCOs) [7], [8] suffer from a nonmonotonic response problem when the frequency band is changed [18]. Therefore, interpolator-based multistage DCOs [13], [18] and a smooth code jumping scheme [17] have been proposed to overcome these problems.

For smart sensing applications, low cost and low power consumption are important. Hence, all-digital PLLs (ADPLLs) are more suitable than analog PLLs, because voltage control schemes face several design challenges in advanced CMOS process at low supply voltages; further, the frequency range for smart sensing applications [8] (<100 MHz) can be easily covered by ADPLLs without necessitating forward body-biased or bulk-driven techniques.

Manuscript received May 5, 2014; revised September 27, 2014 and December 31, 2014; accepted February 20, 2015. Date of publication March 9, 2015; date of current version December 24, 2015. This work was supported by the Ministry of Science and Technology of Taiwan under Grant NSC102-2221-E-194-063-MY3.

The authors are with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 62102, Taiwan (e-mail: wildwolf@cs.ccu.edu.tw; weisiang@s3lab.org; lck@s3lab.org).

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/TVLSI.2015.2407370

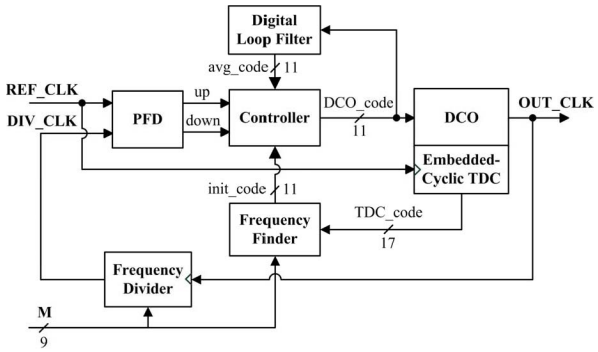

Fig. 1. Block diagram of the proposed ADPLL.

In current ADPLL architectures, the binary search algorithm [12], [13], gear-shifting mechanism [10], and modified bang–bang algorithm [11] have been proposed for reducing the lock-in time. However, several tens of reference clock cycles are still required to achieve lock-in. A DCO control code estimating algorithm has been proposed in [15]. However, the proposed algorithm requires the DCO resolution. A frequency estimation algorithm that employs the period ratio between the reference clock and the DCO output clock to compute the desired DCO control code for generating the target frequency within two cycles has been proposed in [16]. However, the period ratio computed by the frequency counter is an integer number. Therefore, to reduce the frequency error induced by quantization for the period ratio calculation, the multiplication ratio of the ADPLL [16] needs to be maintained at >45.

In this brief, an ADPLL for smart sensing applications employing DVFS power management with two operation modes (0.52/1 V) is presented. The proposed ADPLL can achieve a fast setting time with frequency errors <5% within four clock cycles. Pulse-latch D-type Flip/Flop (DFFs) [19] are used in the ADPLL design for improving the ADPLL performance at low supply voltages.

The rest of this brief is organized as follows. The proposed ADPLL architecture and the proposed frequency estimation algorithm are presented in Section II. Section III describes the circuit implementation of the proposed design. Section IV shows the measurement results of the ADPLL test chip. Finally, the conclusion is drawn in Section V.

II. Proposed ADPLL Architecture

Fig. 1 shows the block diagram of the proposed ADPLL. The ADPLL is composed of a phase and frequency detector (PFD) [12], an ADPLL controller, a digital loop filter [13], a cyclic time-to-digital converter (TDC)-embedded DCO, a frequency finder, and a frequency divider. After the system is reset, the PFD and the frequency divider are stopped. Then the ADPLL controller waits for the frequency finder to compute the initial DCO control code (init_code). Subsequently, the initial DCO control code is applied to the DCO. Then, the DCO, the frequency divider, and the PFD are started at the next rising edge of the reference clock.

Therefore, the phase error is eliminated when the ADPLL returns to the normal phase and frequency acquisition process. However, the limited resolution of the frequency finder causes frequency errors. In addition, the propagation delay time of the frequency divider and the random jitter of the reference clock also cause phase errors. As a result, the proposed ADPLL keeps tracking the phase and frequency of the reference clock (REF_CLK) after the initial frequency estimation process.

The proposed cyclic TDC-embedded DCO uses an interpolator-based fine-tuning stage [18] to maintain a monotonic response between the DCO control code and the output frequency under process, voltage, and temperature (PVT) variations. In addition, the proposed cyclic TDC-embedded DCO can compute the period ratio between the reference clock (REF_CLK) and the DCO output clock (OUT_CLK) as a fixed-point number to reduce the quantization error in the frequency estimation algorithm.

The period of the DCO output clock (OUT_CLK) is a function of an 11-bit DCO control code (DCO_code) and is expressed as P(DCO_code). When the DCO control code is set to zero, the period of the DCO is at its maximum value ($P_{\max}$). Conversely, when the DCO_code is set to 2047, the period of the DCO is at its minimum value ($P_{\min}$). The period ratio between the reference clock (REF_CLK) and the output clock (OUT_CLK) is also a function of DCO_code and is expressed as R(DCO_code). Furthermore, $R_{\max}$ and $R_{\min}$ denote the period ratios when the DCO is operated at $P_{\min}$ and $P_{\max}$, respectively, and can be expressed in

$$R_{\max} = \frac{P_{\text{ref}}}{P_{\min}}, \quad R_{\min} = \frac{P_{\text{ref}}}{P_{\max}} \tag{1}$$

where P_{ref} is the period of the reference clock (REF_CLK).

In [16], R(DCO_code) is determined using a frequency counter. Therefore, the value of R(DCO_code) is an integer number. In this brief, the value of the new function W(DCO_code), which defines as the reciprocal of R(DCO_code), is a fixed-point number. In addition, $W_{\max}$ and $W_{\min}$ can be expressed in

$$W_{\max} = \frac{1}{R_{\min}}, \quad W_{\min} = \frac{1}{R_{\max}}. \tag{2}$$

However, if the value of R(DCO_code) is still determined using a frequency counter, the quantization error in W(DCO_code) is significantly large, as shown in Fig. 2(a). In this brief, we use a cyclic TDC to compute the fixed-point value of R(DCO_code). Therefore, the quantization effects in W(DCO_code) can be significantly reduced, and the W(DCO_code) curve can be approximated as a straight line, as shown in Fig. 2(b). The equation of W(DCO_code) can be expressed in

$$W(\text{DCO\_code}) = \frac{W_{\min} - W_{\max}}{2^{11} - 1} \times \text{DCO\_code} + W_{\max} \tag{3}$$

where DCO_code ranges from 0 to $2^{11} - 1$. Since the frequency multiplication ratio (M) is an input value to the ADPLL, the target period ratio (R_T) is equal to M. Subsequently, the value of the target W(init_code) is equal to $W_T = 1/M$. If the constant value $2^{11} - 1$ in (3) is reduced to 2^{11}, the target DCO control code (init_code) can be computed from (3) and can be further expressed as

$$\text{init\_code} = 2^{11} \times \frac{W_{\max} - W_T}{W_{\max} - W_{\min}}. \tag{4}$$

In the proposed ADPLL, one divider is used to calculate the values of $W_{\max}$, $W_{\min}$, and the initial code of the DCO (init_code) within three clock cycles. As shown in Fig. 3, after the ADPLL is reset, in the first cycle, the state of the ADPLL controller is set to R_MIN and the DCO control code is set to zero such that the proposed cyclic TDC-embedded DCO can be used for calculating the value of $R_{\min}$. In the second cycle, the state is changed to R_MAX and the DCO control code is set to 2047 for calculating the value of $R_{\max}$. Meanwhile, the constant value 2^{19} and $R_{\min}$ are sent to the divider for calculating the $W_{\max}$ value. In the third cycle, the state is changed to PIPE_1 and the constant value 2^{19} and $R_{\max}$ are sent to the divider for calculating the $W_{\min}$ value. In the fourth

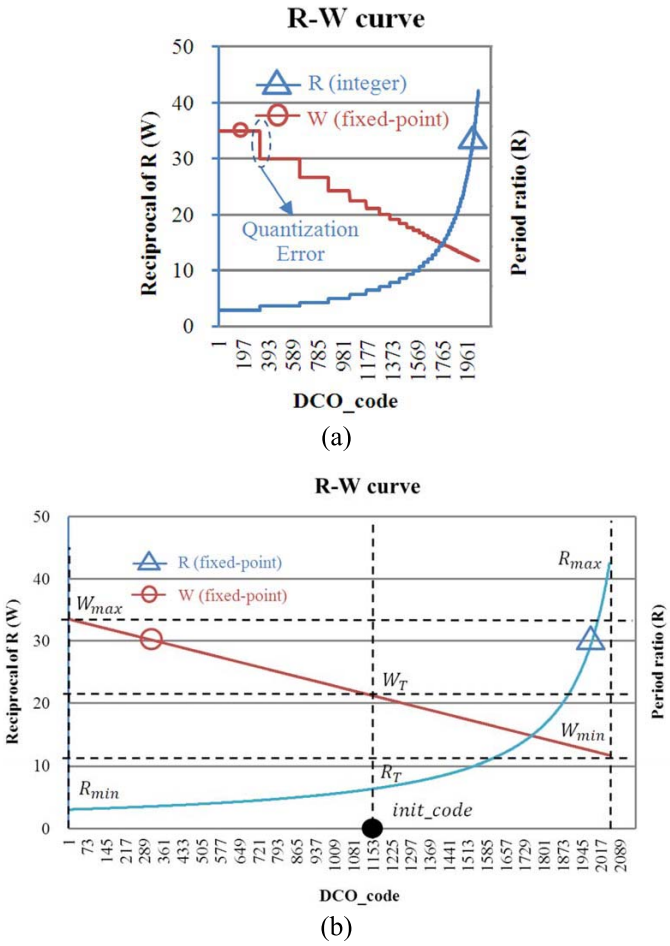

Fig. 2. Relationship of R(DCO_code) and W(DCO_code). (a) R–W curve with integer R. (b) R–W curve with fixed-point R.

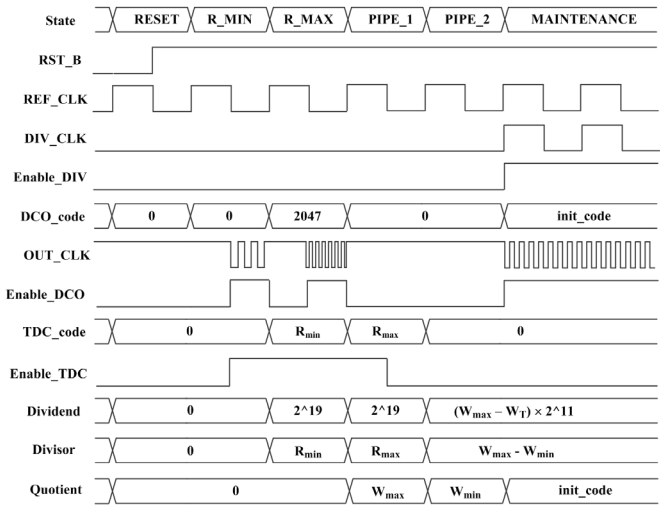

Fig. 3. Timing diagram of the proposed ADPLL in frequency estimation.

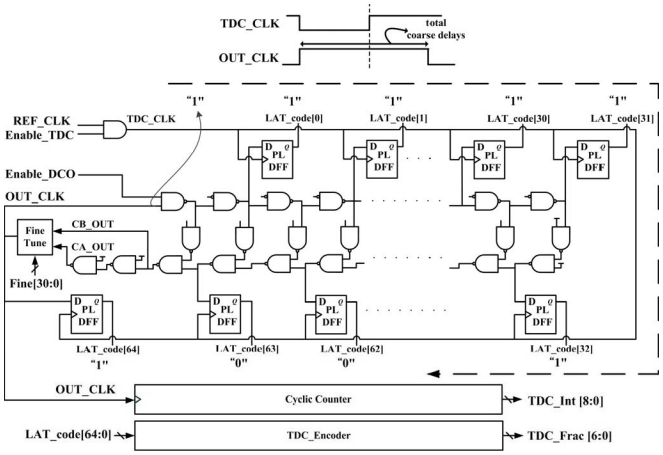

Fig. 4. Proposed cyclic-TDC-embedded DCO.

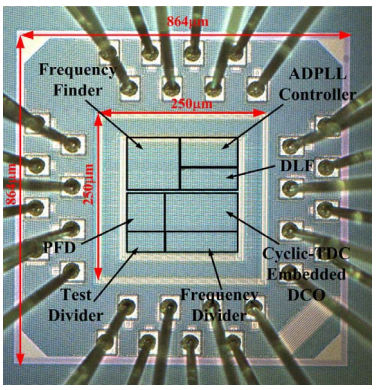

Fig. 5. Chip micrograph.

cycle, the state is changed to PIPE_2 and the initial code of the DCO (init_code) is calculated using (4).

III. Circuit Implementation

Fig. 4 shows the circuit diagram of the proposed cyclic TDC-embedded DCO, which is composed of a coarse-tuning stage, a fine-tuning stage [18], and an embedded cyclic TDC for computing the period ratios ($R_{\max}$ and $R_{\min}$). The coarse-tuning stage of the DCO consists of 64 coarse-tuning delay cells (CDCs), and the coarse-tuning step is the delay time of two NAND gates. The cyclic TDC is composed of 65 pulse-latch DFFs [19] at the output node of the NAND gates, a cyclic counter, and a TDC encoder. The TDC resolution is equal to the delay time of two NAND gates or the coarse-tuning resolution of the DCO. The DCO_Decoder decodes the binary 11-bit DCO_code[10:0] into the coarse-tuning and fine-tuning thermometer codes. The TDC_Encoder encodes the 65-bit thermometer code sampled by the pulse-latch DFFs into a 7-bit TDC fractional number (TDC_Frac[6:0]) and the cyclic counter outputs a 9-bit TDC integer number (TDC_Int[8:0]).

The fine-tuning stage of the DCO [18] is composed of two parallelly connected tristate buffer arrays operating as an interpolator. By adjusting the number of turned-ON tristate buffers in the arrays, the resolution of the fine-tuning stage can be enhanced to 1/32 that of the coarse-tuning step. The fine-tuning stage can maintain a monotonic response between the DCO control code and the output frequency under PVT variations. In addition, the simulated maximum differential nonlinearity is 0.42 least significant bit.

The proposed ADPLL uses a cyclic TDC-embedded DCO to compute the fixed-point values of $R_{\max}$ and $R_{\min}$ for calculating the target DCO control code (init_code). As shown in Figs. 3 and 4, at the initial state, the state of the ADPLL controller is RESET and both Enable_TDC signal and Enable_DCO signal are set to zero to gate the TDC_CLK signal; further, and all the CDCs are enabled and initialized. In the next REF_CLK cycle, the state of the ADPLL controller is changed to R_MIN and both Enable_TDC signal and Enable_DCO signal are pulled up at the negative edge of the REF_CLK. The Enable_DCO signal is pulled up for one half of the REF_CLK period. In the next positive edge of the REF_CLK, the node voltage values of the CDCs are sampled by the pulse-latch DFFs. Then, the Enable_DCO signal is set to zero again to initialize all the CDCs for the subsequent $R_{\max}$ calculation.

As shown in (1), the period ratio between the reference clock (REF_CLK) and the maximum output period of the output clock (OUT_CLK) is $R_{\min}$. To pipeline the initial code calculation procedure within four clock cycles, the Enable_DCO signal is pulled up for one half of the REF_CLK period. As a result, the output of the cyclic TDC should be multiplied by 2 and then denoted as TDC_code[16:0].

Fig. 4 shows the $R_{\min}$ calculation in the case when the half period of the reference clock (REF_CLK) is less than the delay time of the full coarse-tuning delay line. When both Enable_TDC signal and Enable_DCO signal are pulled up at the negative edge of REF_CLK, the DCO begins to oscillate, and the 1 on the OUT_CLK node is propagated to the NAND gate delay chain. After the half cycle of REF_CLK, the node voltage values of the CDCs are sampled by the pulse-latch DFFs as LAT_code[63:0]. The TDC_Encoder encodes the number of 1's in the LAT_code[63:0] as a 7-bit TDC fractional number (TDC_Frac[6:0]).

As shown in Fig. 3, when the state of the ADPLL controller is changed to R_MAX, the DCO control code is set to 2047 for calculating the value of $R_{\max}$. Therefore, all the CDCs are disabled and OUT_CLK triggers the cyclic counter to obtain a 9-bit TDC integer number (TDC_Int[8:0]). At the positive edge of TDC_CLK, the TDC_code[16:0] = {TDC_Int[8:0], 7'h0, 1'b0} is outputted as the period ratio $R_{\max}$.

The frequency finder, the digital loop filter, and the ADPLL controller are designed with hardware description language. After logic synthesis, these circuits are mapped to logic gates, and the other blocks are also implemented with standard cells. In addition, pulse-latch DFFs [19] are employed in the frequency divider and the PFD design. In the proposed PFD [12], the dead zone of the PFD is reduced from 107 to 26 ps at 0.52 V.

IV. Experimental Results

The proposed ADPLL is implemented using a standard performance 90-nm CMOS process. The micrograph of the test chip is shown in Fig. 5. The active area is 250 × 250 μm^2 and the chip area including the I/O pads is 864 × 864 μm^2. The output frequency ranges from 60 to 600 MHz at 1 V, and from 30 to 120 MHz at 0.52 V. The power consumption of the proposed ADPLL is 0.92 mW at (1 V, 600 MHz) and 37 μW at (0.52 V, 120 MHz).

Fig. 6 shows the measured jitter histogram of the output clock at (0.52 V, 120 MHz). The peak-to-peak (P_K–P_K) jitter is 155 ps, and the root mean square (rms) jitter is 26.8 ps. Due to the speed limitations of the I/O pad, the output clock is divided by 2 before it is outputted to the I/O pad. The P_K–P_K jitter at (1 V, 600 MHz) is 102 ps, and the rms jitter is 13.7 ps. At low supply voltages, the DCO resolution and the PFD dead zone worsen, and, therefore, the jitter performance of the ADPLL at 0.52 V is worse than that at 1 V. Moreover, the maximum output frequency of the ADPLL

TABLE I
PERFORMANCE COMPARISONS

Parameter	Proposed	[6] JSSC'12	[5] TCAS-I'11	[8] A-SSCC'12	[9] TCAS-II'12	[21] TCAS-II'13	[16] TCAS-II'10
Category	ADPLL	Analog PLL	Analog PLL	ADPLL	ADPLL	ADPLL	ADPLL
Process	90nm	0.13µm	90nm	40nm	90nm	28nm	0.18µm
Power Supply	0.52V/1.0V	0.5V	0.5V	0.5V	0.5V	1.0V	1.8V
Area (mm^2)	0.065	0.074	0.074	0.049	0.022	0.00234	0.075
Input Frequency (MHz)	< 20	1.8432	25 ~ 140	1.0	6 ~ 45	50	0.22~8
Output Frequency (MHz)	30~120 @0.52V 60~600 @1.0V	400 ~ 433	400 ~ 2240	10 ~ 100	96 ~ 720	83 ~ 2000	223 ~ 446
Multiplication Factor	2 ~ 128	206 ~ 236	16	10 ~ 100	16	2 ~ 40	45 ~ 128
DCO Resolution or VCO gain	20.9ps@0.52V 7.3ps@1.0V	300 ~ 450 MHz/V	1200 MHz/V	43.9ps	1ps	N/A	8.8ps @446MHz
$Jitter_{RMS}$	26.8ps@ 0.52V,120MHz 13.7ps@ 1.0V,600MHz	5.5ps @433MHz	9.62ps @400MHz 2.22ps @2.24GHz	44.2ps @100MHz	50.0ps @96MHz 13.3ps @720MHz	3.0ps @2.0GHz	9.4ps @446MHz
Jitter/cycle (%)	0.32@ 0.52V,120MHz 0.82@ 1.0V,600MHz	0.24 @433MHz	0.38 @400MHz 0.497 @2.24GHz	0.44 @100MHz	0.48 @96MHz 0.95 @720MHz	0.6 @2.0GHz	0.42 @446MHz
Power Consumption	37µW@ 0.52V,120MHz 0.92mW@ 1.0V,600MHz	0.44mW @433MHz	0.5mW @400MHz 2.08mW @2.24GHz	47µW @100MHz	0.57mW @720MHz	0.64mW @2.0GHz	14.5mW @446MHz
Lock-in Time (frequency error <5%)	4 cycles (0.8µs)	166 cycles (90µs)	210 cycles (1.5µs)	50 cycles (50µs)	N/A	50 cycles (1.0µs)	2 cycles (0.5µs)

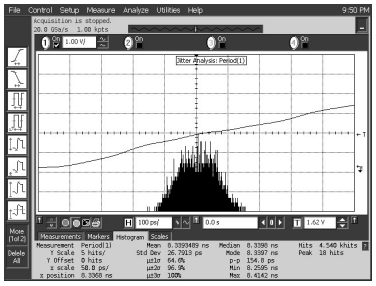

Fig. 6. Measured jitter histogram at 0.52 V and 120 MHz.

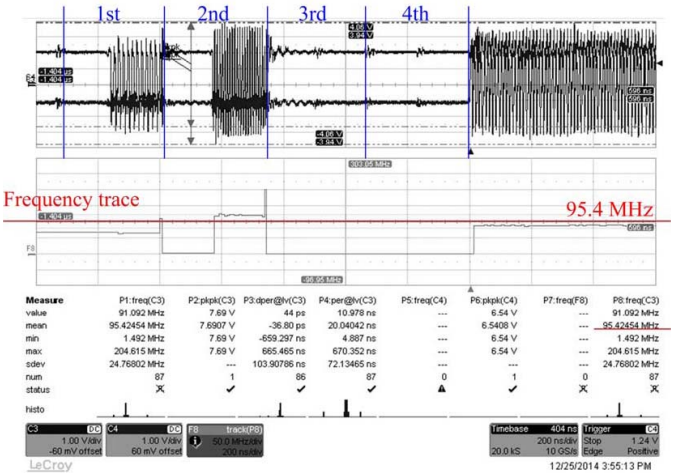

Fig. 7. Measured lock-in process of the proposed ADPLL.

also reduces at 0.52 V. Fig. 7 shows the measured lock-in process of the proposed ADPLL at 1 V. The reference clock is 3 MHz, the frequency multiplication ratio (M) is 32, and thus, the output frequency is 96 MHz. After four clock cycles, the frequency error is <5%.

Table I lists the performance comparisons with other PLLs. Although analog low-voltage PLLs [5], [6] have a higher output frequency at low supply voltages, a high VCO gain makes the VCO susceptible to supply noise and noise induced by nearby digital circuits. As compared with digital low-voltage PLLs [8], the proposed ADPLL has better jitter performance and lower power consumption. As compared with other ADPLLs [13], [16], the proposed ADPLL has lower power consumption. As compared with the ADPLL [16], the proposed ADPLL improves the accuracy of the period ratio computation and can achieve a fast setting time at small frequency errors within four clock cycles. In addition, as compared with another fast-locking ADPLL [21], which requires ∼50 cycles lock-in time, the proposed ADPLL has a faster settling time. As compared with the near-threshold ADPLL [20], the bootstrapped DCO can improve the maximum output frequency with higher design complexity. The maximum output frequency of the proposed ADPLL at 0.52 V is enough for the smart sensing applications with DVFS power management [8] (<100 MHz).

V. CONCLUSION

In this brief, a fast lock-in ADPLL for smart sensing applications employing DVFS power management has been proposed. The proposed frequency estimation algorithm can use the period ratios calculated by the cyclic TDC to compute the initial DCO control code for achieving a fast setting time within four clock cycles. In addition, the interpolator-based fine-tuning stage can maintain a monotonic response of the DCO either at 1 or 0.52 V. Pulse-latch DFFs are applied to the design of the frequency divider and PFD to improve the performance of the ADPLL at low supply voltages. Furthermore, the proposed ADPLL has good portability

over different processes. Therefore, the proposed ADPLL is suitable for battery-powered energy-efficient processing platforms employing DVFS schemes.

REFERENCES

[1] J. Kwong and A. P. Chandrakasan, "An energy-efficient biomedical signal processing platform," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1742–1753, Jul. 2011.

[2] J.-Y. Yu *et al.*, "A 0.5 V 4.85 Mbps dual-mode baseband transceiver with extended frequency calibration for biotelemetry applications," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Nov. 2008, pp. 293–296.

[3] C.-T. Lu, H.-H. Hsieh, and L.-H. Lu, "A low-power quadrature VCO and its application to a 0.6-V 2.4-GHz PLL," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 4, pp. 793–802, Apr. 2010.

[4] S.-A. Yu and P. Kinget, "A 0.65-V 2.5-GHz fractional-N synthesizer with two-point 2-Mb/s GFSK data modulation," *IEEE J. Solid-State Circuits*, vol. 44, no. 9, pp. 2411–2425, Sep. 2009.

[5] K.-H. Cheng, Y.-C. Tsai, Y.-L. Lo, and J.-S. Huang, "A 0.5-V 0.4–2.24-GHz inductorless phase-locked loop in a system-on-chip," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 5, pp. 849–859, May 2011.

[6] W.-H. Chen, W.-F. Loke, and B. Jung, "A 0.5-V, 440-μW frequency synthesizer for implantable medical devices," *IEEE J. Solid-State Circuits*, vol. 47, no. 8, pp. 1896–1907, Aug. 2012.

[7] J. A. Tierno, A. V. Rylyakov, and D. J. Friedman, "A wide power supply range, wide tuning range, all static CMOS all digital PLL in 65 nm SOI," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 42–51, Jan. 2008.

[8] Y. Hiraku, I. Hayashi, H. Chung, T. Kuroda, and H. Ishikuro, "A 0.5 V 10 MHz-to-100 MHz 0.47 μW/MHz power scalable AD-PLL in 40 nm CMOS," in *Proc. IEEE Asian Solid-State Circuits Conf.*, Nov. 2012, pp. 33–36.

[9] K.-H. Cheng, J.-C. Liu, and H.-Y. Huang, "A 0.6-V 800-MHz all-digital phase-locked loop with a digital supply regulator," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 12, pp. 888–892, Dec. 2012.

[10] R. B. Staszewski and P. T. Balsara, "All-digital PLL with ultra fast settling," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 2, pp. 181–185, Feb. 2007.

[11] C.-C. Hung and S.-I. Liu, "A 40-GHz fast-locked all-digital phase-locked loop using a modified bang-bang algorithm," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 58, no. 6, pp. 321–325, Jun. 2011.

[12] C.-C. Chung and C.-Y. Lee, "An all-digital phase-locked loop for high-speed clock generation," *IEEE J. Solid-State Circuits*, vol. 38, no. 2, pp. 347–351, Feb. 2003.

[13] C.-C. Chung and C.-Y. Ko, "A fast phase tracking ADPLL for video pixel clock generation in 65 nm CMOS technology," *IEEE J. Solid-State Circuits*, vol. 46, no. 10, pp. 2300–2311, Oct. 2011.

[14] H.-J. Hsu and S.-Y. Huang, "A low-jitter ADPLL via a suppressive digital filter and an interpolation-based locking scheme," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 1, pp. 165–170, Jan. 2011.

[15] G. Yu, Y. Wang, H. Yang, and H. Wang, "Fast-locking all-digital phase-locked loop with digitally controlled oscillator tuning word estimating and presetting," *IET Circuits, Devices Syst.*, vol. 4, no. 3, pp. 207–217, May 2010.

[16] C.-T. Wu, W.-C. Shen, W. Wang, and A.-Y. Wu, "A two-cycle lock-in time ADPLL design based on a frequency estimation algorithm," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 57, no. 6, pp. 430–434, Jun. 2010.

[17] P.-Y. Chao, C.-W. Tzeng, S.-Y. Huang, C.-C. Weng, and S.-C. Fang, "Process-resilient low-jitter all-digital PLL via smooth code-jumping," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 12, pp. 2240–2249, Dec. 2013.

[18] D. Sheng and J.-C. Lan, "A monotonic and low-power digitally controlled oscillator with portability for SoC applications," in *Proc. IEEE 54th Int. Midwest Symp. Circuits Syst.*, Aug. 2011, pp. 1–4.

[19] W.-H. Sung, M.-C. Lee, C.-C. Chung, and C.-Y. Lee, "Ultra-low voltage implicit multiplexed differential flip-flop with enhanced noise immunity," *Electron. Lett.*, vol. 48, no. 23, pp. 1452–1454, Nov. 2012.

[20] Y.-C. Ho, Y.-S. Yang, C.-C. Chang, and C.-C. Su, "A near-threshold 480 MHz 78 μW all-digital PLL with a bootstrapped DCO," *IEEE J. Solid-State Circuits*, vol. 48, no. 11, pp. 2805–2814, Nov. 2013.

[21] S. Höppner, S. Haenzsche, G. Ellguth, D. Walter, H. Eisenreich, and R. Schüffny, "A fast-locking ADPLL with instantaneous restart capability in 28-nm CMOS technology," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 60, no. 11, pp. 741–745, Nov. 2013.