A 0.5V/1.0V Fast Lock-In ADPLL for DVFS Battery-Powered Devices

Ching-Che Chung[1], Duo Sheng[2], and Wei-Siang Su [1]
[1] Department of Computer Science and Information Engineering, National Chung Cheng University,
No. 168 University Rd., Min-Hsiung, Chia-Yi, Taiwan
[2] Department of Electrical Engineering, Fu Jen Catholic University,
No. 510 Chung-Cheng Rd. Hsin-Chung, Taipei, Taiwan.
Email: wildwolf@cs.ccu.edu.tw

Abstract **— In this paper, a 0.5V/1.0V low-power all-digital phase-locked loop (ADPLL) for battery-powered devices with a dynamic voltage and frequency scaling (DVFS) scheme is presented. The proposed frequency estimation algorithm with a fine-resolution monotonic response digitally controlled oscillator (DCO), can quickly calculate the target control code for the DCO, and thus, the ADPLL can achieve a fast lock time in four clock cycles. The proposed ADPLL is implemented in a standard performance 65nm CMOS process with standard cells. The power consumption is 52.69μW at 600MHz with a 0.5V power supply and is 1.26mW at 1.28GHz with a 1.0V supply.**

I. INTRODUCTION

In recent years, biomedical electronic applications, such as biological signal monitoring devices, implantable medical devices [1], and wireless body sensors [2] become more and more popular now. In these battery-powered systems, low energy is a primary concern to increase the system operating time. Therefore, power management is an important issue for designing these devices. Dynamic voltage and frequency scaling (DVFS) serves an effective means to reduce the dynamic power consumption of the system. Moreover, the duty-cycle control of the power switch can further reduce the standby power consumption of the system.

In the system-on-a-chip (SoC), there are several phase-locked loops (PLLs) and delay-locked loops (DLLs) to provide different clock sources for different modules. However, conventional charge-pump based phase-locked loops (CP-PLLs) are often not workable at a low voltage. Moreover, the CP-PLLs usually take a long lock-in time, and thus, they are not possible to be turned off for reducing the standby power consumption. When the system is switched to the sleeping mode, the continuous operating PLLs often dominate the standby power consumption of the system. As a result, the PLLs which can both operate at a low voltage and have a fast lock-in time are demanded in these applications.

The forward body bias (FBB) [1] and reverse short channel effect (RSCE) [3] are two major techniques to increase the driving strength of the analog circuits at a low voltage. Thus CP-PLLs [1], [6] usually use these techniques to design their PLLs. However, the gain (K_{VCO}) of the voltage controlled oscillator (VCO) becomes very large due to the restricted voltage headroom at a low voltage. Even a multi-band VCO architecture is adopt in CP-PLLs, the K_{VCO} is still very large. For instance, the K_{VCO} of the CP-PLL [6] is 1200MHz/V, which means the VCO output frequency is very sensitive to the little noise on the control voltage. In addition, the FBB [1] technique needs a triple-well process technology, and thus, it increases the cost of chip fabrication. In addition, the leakage current problem of the charge-pump also increases the design challenges of the CP-PLLs in advanced CMOS processes.

All-digital phase-locked loops (ADPLLs) [7]-[10],[13],[14] usually achieve a relatively fast lock-in time than the CP-PLLs and can be easily integrated with other digital circuits. Therefore, ADPLL is suitable for biomedical electronic applications. The binary search scheme [7] is widely adopted to reduce the lock-in time of the ADPLLs. The time-to-digital converter (TDC)-based ADPLL [9] uses the delay cells of the DCO to quantize the reference clock period information to further reduce the lock-in time to seven cycles. However, the ADPLL [9] has a poor DCO resolution which depends on the inverter delay, and thus, it is not suitable for high-speed clock generation with a low supply voltage.

In [10], a frequency estimation algorithm is proposed for the ADPLL controller to quickly find out the target DCO control code, and then, it can achieve a lock-in time with two clock cycles. However, the proposed frequency estimation algorithm needs a fine-resolution high-linearity DCO with the monotonic response. Moreover, the ADPLL [10] also requires a large frequency multiplication ratio to reduce the quantization effect of the frequency counter, or the target DCO control code calculated using the proposed equation will cause a large frequency error after two clock cycles. In addition, the ADPLL [10] uses a frequency counter to obtain the require cycle count information. Thus, the cycle time ratios between the reference clock and the DCO are integer numbers which cause considerable calculation errors in the proposed equation. Moreover, the proposed tri-state inverter-based DCO does not have a fine-tuning stage, and thus, it cannot achieve the wide-range operation with both a fine resolution and a high linearity. As a result, the frequency range of the DCO is very limited, and thus, it is not suitable for biomedical electronic applications with DVFS scheme.

Although the transition time and the propagation delay of the logic gates are increased with a lower supply voltage, but most logic gates of the standard cell library can still work correctly at a low supply voltage. However, the sequential elements (i.e. D-type flip-flops, DFFs) often have unacceptable setup time and hold time margins and a large clock-to-Q delay at a low supply voltage. In the ADPLL, the DFFs of the frequency divider operate at the maximum frequency of the DCO, and thus, a smaller clock-to-Q delay with narrower setup time and hold time margins are needed. In addition, the DFFs of the TDC require a smaller time margins for reducing the sample error. The DFFs of the phase and frequency detector (PFD) also needs a smaller time margins for reducing the dead zone. As a result, the design of the DFFs at a low supply voltage is very important. The FBB technique with a true single phase clock (TSPC) DFF [1], [4] or the pulse-latch DFF [5] can help to build up the DFF for the low-voltage ADPLL.

In this paper, we proposed an ADPLL for battery-powered devices with DVFS scheme. The proposed frequency estimation algorithm with an embedded cyclic TDC can quickly calculate the target DCO control code with high precision, and thus, the proposed ADPLL can achieve a fast lock time in four clock cycles. The rest of the paper is organized as follows: Section II presents the proposed ADPLL architecture and the frequency estimation algorithm. Section III discusses the circuit implementation. Section IV shows experimental results. Finally, Section V concludes with a summary.

This work was supported in part by the National Science Council of Taiwan, under Grant NSC101-2221-E-194-063.

978-1-4673-4436-4/13/$31.00 ©2013 IEEE

II. ARCHITECTURE OVERVIEW

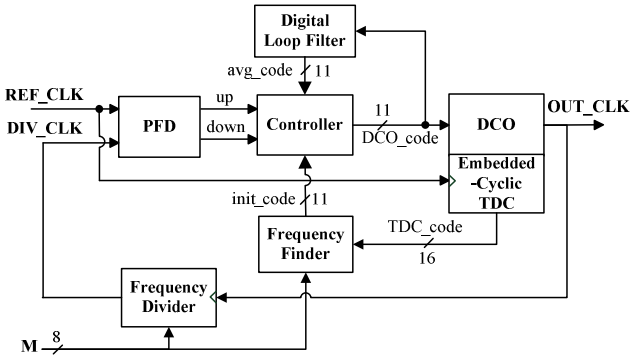

FIGURE 1. THE PROPOSED ADPLL.

The block diagram of the proposed ADPLL is shown in Fig. 1. The ADPLL is composed of a phase and frequency detector (PFD), an ADPLL controller with a digital loop filter (DLF), a frequency finder (FF), a monotonic low-power cyclic TDC-embedded DCO, and a frequency divider. After system is reset, the PFD and the frequency divider are stopped waiting for the frequency finder to calculate the initial DCO control code (*init_code*) for the ADPLL controller. Then, the PFD and the frequency divider are enabled, and the proposed ADPLL can achieve lock in four clock cycles. Subsequently, the ADPLL controller updates the DCO control code (DCO_code) according to the PFD's output to keep tracking the phase and frequency of the reference clock. The digital loop filter [14] is applied to produce the baseline DCO control code (avg_code) for reducing the reference clock jitter effects.

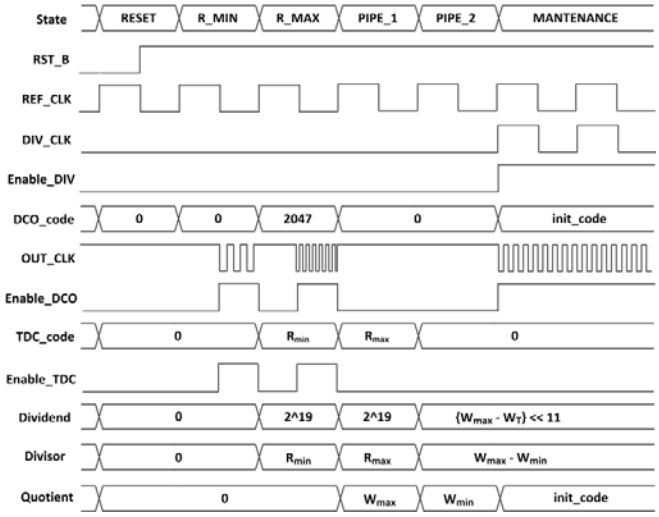

FIGURE 2. TIMING DIAGRAM OF FREQUENCY ESTIMATION.

The timing diagram of the proposed frequency estimation algorithm is shown in Fig. 2. The period of the DCO output clock (OUT_CLK) is a function of the DCO control code, and is named as *P(code)*. When the DCO control code is set to zero, the period of the DCO is at its maximum value (P_{max}), as shown in Fig. 3(a). Oppositely, when the DCO control 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 DCO clock (OUT_CLK) is also a function of the DCO control code, and is named as *R(code)*. The R_{max} and R_{min} mean that the period ratios when the DCO operates at P_{min} and P_{max}, respectively. The value of R_{max} is equal to P_{ref}/P_{min}, and R_{min} is equal to P_{ref}/P_{max}, where P_{ref} is the period of the reference clock. In [10], the ratio *R(code)* is obtained by using a frequency counter. Therefore, the value of the *R(code)* will be an integer number, and it has quantization effects, as shown in Fig. 3(a). If we define a new function *W(code)* which is the reciprocal of the *R(code)*, and then, the value of the *W(code)* will be a fixed-point number. However, if the value of *R(code)* is still obtained by a frequency counter, the quantization error of *W(code)* will be very large, as shown in Fig. 3(b).

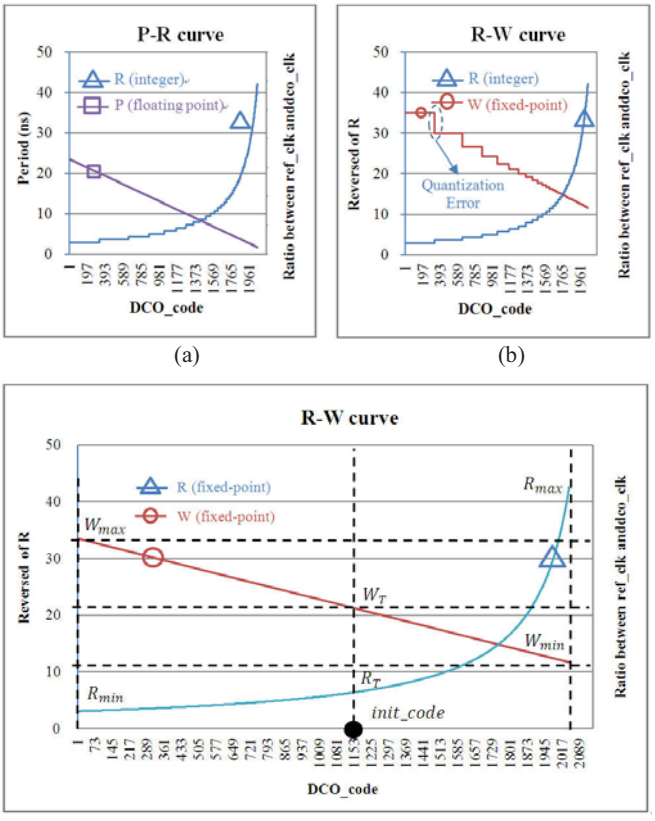

FIGURE 3. THE RELATIONSHIP OF ***P***, ***R***, AND ***W***.

In this paper, we uses a cyclic TDC to obtain the fixed-point value of the *R(code)*. Thus, the quantization effects of *W(code)* can be significantly reduced, as shown in Fig. 3(c). In Fig. 3(c), W_{max} is equal to $1/R_{min}$, and W_{min} is equal to $1/R_{max}$, and then, the *W(code)* curve will be a straight line. Therefore, the equation of *W(code)* can be expressed as Eq. 1. Since the frequency multiplication factor (*M*) is an input value, the target period ratio (R_T) is equal to M ($=P_{ref}/P_T$), where P_T is the target DCO period. Then, the value of the target *W(init_code)* is equal to $W_T=1/M$. If the constant value (2^{11}-1) in the Eq. 1 is reduced as 2^{11}, the target DCO control code (*init_code*) can be found using Eq. 2.

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

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

In the proposed ADPLL, one divider is used to calculate the value of W_{max}, W_{min}, and the initial code of the DCO (*init_code*) in three clock cycles. In Fig. 2, after system is reset, in the first cycle, the state of the ADPLL controller is **R_MIN**, and in this cycle, the DCO control code is set to zero to use the proposed cyclic TDC for calculating the value of R_{min}. In the second cycle, the state is changed to **R_MAX**, 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}. In the third cycle, the state is

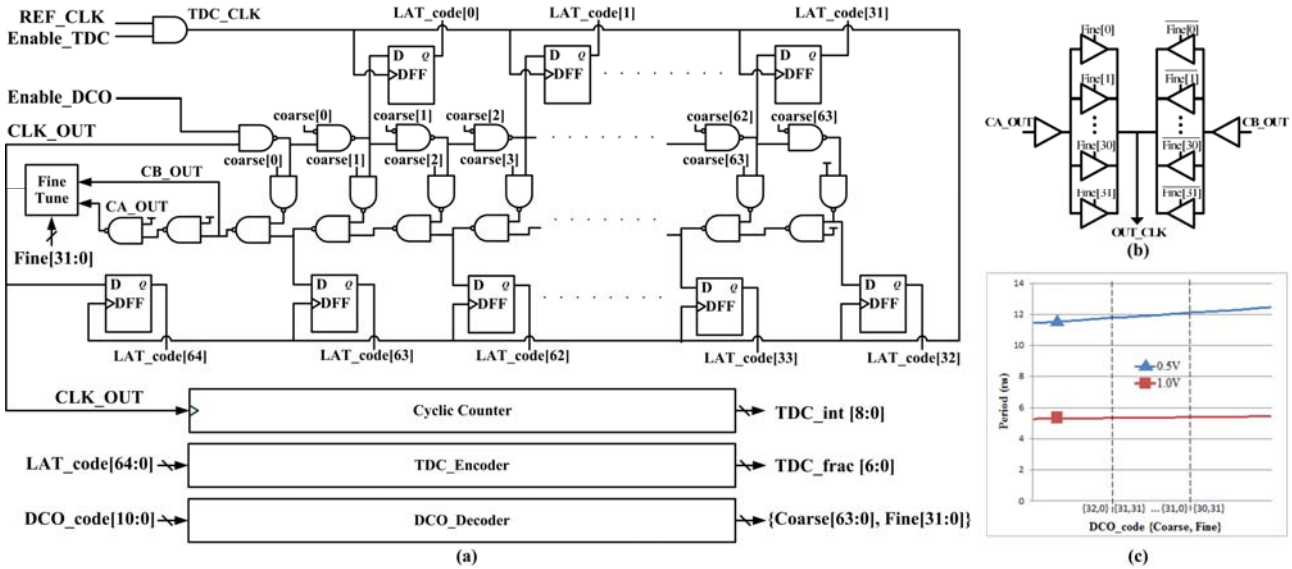

FIGURE 4. THE PROPOSED DCO WITH EMBEDDED CYCLIC TDC.

changed to **PIPE_1**, the constant value 2^{19} and R_{max} are sent to the divider for calculating the W_{min}. In the fourth cycle, the state is changed to **PIPE_2**, and the initial code of the DCO (*init_code*) is calculated using Eq. 2.

In the ADPLL [10], three DCOs are used for calculating the required parameters. However, these DCOs may have on-chip variations (OCVs) especially in advanced CMOS process. In addition, the integer values of R_{min} and R_{max} are used in the ADPLL [10] for calculating the target DCO control code which results in large quantization errors. Therefore, the ADPLL [10] still have a large frequency error after two clock cycles. The proposed ADPLL uses a monotonic low-power cyclic TDC-embedded DCO to avoid on-chip variations, and we use fixed-point values of R_{max} and R_{min} with the proposed frequency estimation algorithm for calculating an accurate target DCO control code (*init_code*). As a result, the proposed ADPLL can achieve a relatively small frequency error after four clock cycles.

III. CIRCUIT IMPLEMENTATION

The proposed DCO with an embedded cyclic TDC is shown in Fig. 4(a). The coarse-tuning stage [11] is consisted of 64 coarse-tuning delay cells (CDCs). The coarse-tuning resolution of the proposed DCO is two NAND gate delay times. Moreover, the unused CDCs can be gated for reducing the power consumption of the DCO at high frequency operation. The fine-tuning stage of the DCO is shown in Fig. 4(b). The fine-tuning stage [12] is composed of two parallel connected tri-state buffer arrays operating as an interpolator. When more left-hand side tri-state buffers are turned on, the output clock is more close to the CA_OUT, and when more right-hand side tri-state buffers are turned on, the output clock is more close to the CB_OUT. In addition, the timing difference between CA_OUT and CB_OUT is one coarse-tuning resolution. Therefore, the proposed fine-tuning stage can always provide a total delay tuning range equal to one coarse-tuning resolution under process, voltage, and temperature (PVT) variations.

In the proposed frequency estimation algorithm, a high resolution DCO with the monotonic response is required, thus the proposed fine-tuning stage can overcome this problem and achieves a fine resolution. Fig. 4(c) shows the simulation results of the proposed DCO. The proposed DCO keeps the monotonic response when the DCO control code switches cross over different coarse-tuning control codes. The coarse-tuning resolution is about 330.5ps and 163.9ps at 0.5V and 1.0V, respectively. In addition, the fine-tuning resolution is 10.7ps and 5.1ps at 0.5V and 1.0V, respectively.

After system is reset, the state of the ADPLL controller is **R_MIN**, and in this cycle, the DCO control code is set to zero for calculating the value of R_{min}. The Coarse[63:0] is set to 64'h0 to turn on the full delay line. Then, the TDC is enabled at the negative edge of the reference clock, and the "0" begins to pass through the delay line. At the next positive reference clock edge, the DFFs captures the output of the CDCs and the TDC encoder encodes them as the fractional TDC control code (TDC_frac[6:0]). If the half cycle time of the reference clock is longer than the delay time of the full delay line, a cyclic counter is triggered, and the integer TDC control code (TDC_int[8:0]) is recorded. Similarly, at the **R_MAX** state, the DCO control code is set to *2047* for calculating the value of R_{max}. Then the fixed-point TDC control code (TDC_code) is output to the proposed frequency finder. Moreover, the conventional DFFs have unacceptable setup time and hold time margins and a large clock-to-Q delay at a low supply voltage. Therefore, we use the pulse-latch DFFs to enhance the resolution of the TDC, and the pulse-latch DFFs are also applied to the frequency divider and the PFD circuit [13].

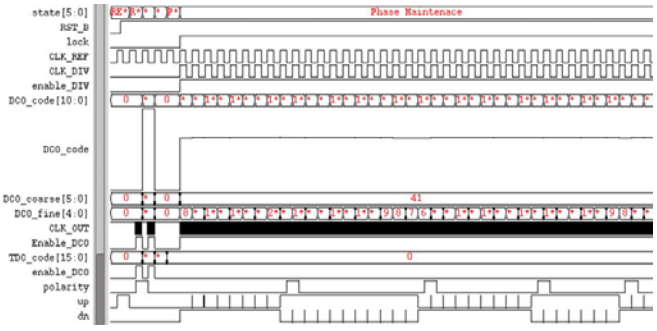

FIGURE 5. SIMULATION RESULTS OF THE PROPOSED ADPLL.

IV. EXPERIMENTAL RESULTS

The proposed ADPLL is implemented in a standard performance 65nm CMOS process with standard cells. The simulation result of the proposed ADPLL at 1.0V with M=128 is shown in Fig. 5. After system is reset, the proposed frequency finder computes the target

TABLE I. PERFORMANCE COMPARISONS

Parameter	Proposed	[1] JSSC'12	[6] TCAS-I'11	[7] JSSC'11	[8] TCAS-I'12	[9] JSSC'03	[10] TCAS-II'10
Process	65nm	0.13μm	90nm	65nm	90nm	0.65μm	0.18μm
Category	All-Digital PLL	Analog PLL	Analog PLL	All-Digital PLL	All-Digital PLL	All-Digital PLL	All-Digital PLL
Supply Voltage	0.5V/1.0V	0.5V	0.5V	1.0V	1.0V/1.2V	5V	1.8V
Input Frequency (MHz)	5	1.8432	280	0.036 ~12.5	60	0.011 ~ 0.339	0.22~8
Output Frequency (MHz)	45~600@0.5V 100~1280@1.0V	400 ~ 433	400 ~ 2240	90 ~ 527	4080	0.045 ~ 61.3	222.6 ~ 445.8
Time Resolution	10.7ps@0.5V 5.1ps@1.0V	450 MHz/V	1200 MHz/V	16.2 ps	20 kHz/LSB	170 ps	8.8 ps
Multiplication Factor	2 ~ 256	206 ~ 236	8	16 ~ 5600	64	4 ~ 1022	45~128
Power Consumption	52.69 μW @(0.5V,600MHz) 1.26 mW @(1.0V,1.28GHz)	440 μW @433 MHz	2.08 mW @2.24GHz	1.81mW @520MHz	8.48mW@1.0V 10.08mW@1.2V	N/A	14.5 mW @446MHz
Lock-in Time	4 cycles	165 cycles	N/A	N/A	< 45 cycles	7 cycles	2 cycles

DCO control code in four clock cycles. After that, the frequency divider and the PFD are turned on, and the proposed ADPLL keeps tracking the phase and frequency of the reference clock. Table I lists the comparisons with prior studies. Although the ADPLL [10] can achieve lock in two clock cycles, it still has a large frequency error due to the on-chip variations of the three DCOs and the quantization effects of the frequency counter. The proposed ADPLL has lowest power consumption and a relatively fast lock-in time. As a result, it is suitable for biomedical electronic applications with DVFS scheme.

V. CONCLUSION

In this paper, the proposed frequency estimation algorithm can use the period ratio information calculated from the cyclic TDC to compute the target DCO control code in four clock cycles. In addition, the proposed ADPLL can work at a low supply voltage with ultra-low-power consumption. The proposed ADPLL also has good portability over different processes. Therefore, the proposed ADPLL is very suitable for battery-powered SoC with DVFS scheme.

ACKNOWLEDGMENT

The authors would like to thank their colleagues in the Silicon Sensor and System (S3) Laboratory of National Chung Cheng University for many fruitful discussions. The EDA tools supported by National Chip Implementation Center (CIC) are acknowledged as well.

REFERENCES

[1] Wu-Hsin Chen, et al., "A 0.5-V, 440-μW frequency synthesizer for implantable medical devices, " *IEEE Journal of Solid-State Circuits*, vol. 47, no. 8, pp. 1896-1907, Aug. 2012.

[2] Wei-Hao Sung, Jui-Yuan Yu, and Chen-Yi Lee, "A robust frequency tracking loop for energy-efficient crystalless WBAN systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 58, no. 10, pp. 637-641, Oct. 2011.

[3] Tae-Hyoung Kim, et al., "Utilizing reverse short-channel effect for optimal subthreshold circuit design," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 15, no. 7, pp. 821-829, Jul. 2007.

[4] Seungsoo Kim, and Hyunchol Shin, "An E-TSPC divide-by-2 circuit with forward body biasing in 0.25μm CMOS," *IEEE Microwave and Wireless Components Letters*, vol. 19, no. 10, pp. 656-658, Oct. 2009.

[5] Chulwoo Kim, and Sung-Mo (Steve) Kang, "A low-swing clock double-edge triggered flip-flop, " *IEEE Journal of Solid-State Circuits*, vol. 37, no. 5, pp. 648-652, May 2002.

[6] Kuo-Hsing Cheng, et al., "A 0.5-V 0.4-2.24-GHz inductorless phase-locked loop in a system-on-chip, " *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 5, pp. 849-859, May. 2011.

[7] Ching-Che Chung and Chiun-Yao Ko, "A fast phase tracking ADPLL for video pixel clock generation in 65nm CMOS technology, " *IEEE Journal of Solid-State Circuits,* vol. 46, no. 10, pp. 2300-2311, Oct. 2011.

[8] Ja-Yol Lee, et al., "A 4-GHz all digital PLL with low-power TDC and phase-error compensation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 8, pp. 1706-1719, Aug. 2012.

[9] Takamoto Watanabe and Shigenori Yamauchi, "An all-digital PLL for frequency multiplication by 4 to 1022 with seven-cycle lock time, " *IEEE Journal of Solid-State Circuits*, vol. 38, no. 2, pp. 198-204, Feb. 2003

[10] Chia-Tsun Wu, et al., "A two-cycle lock-in time ADPLL design based on a frequency estimation algorithm, " *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 6, pp. 430-434, Jun. 2010.

[11] Rong-Jyi Yang and Shen-Iuan Liu, "A 40-550 MHz harmonic-free all-digital delay-locked loop using a variable SAR algorithm, " *IEEE Journal of Solid-State Circuits*, vol. 42, no. 2, pp. 361-373, Feb. 2007.

[12] Duo Sheng and Jhih-Ci Lan, "A monotonic and low-power digitally controlled oscillator with portability for SoC applications," *in Proceedings of IEEE International Midwest Symposium on Circuits and Systems (MWSCAS),* Aug. 2011.

[13] Ching-Che Chung and Wei-Cheng Dai, "A referenceless all-digital fast frequency acquisition full-rate CDR circuit for USB 2.0 in 65nm CMOS technology, " *in Proceedings of International Symposium on VLSI Design, Automation, and Test (VLSI-DAT)*, Apr. 2011, pp. 217-220.

[14] Hsuan-Jung Hsu and Shi-Yu Huang, "A low-jitter ADPLL via a suppressive digital filter and an interpolation-based locking scheme," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 1, pp. 165-170, Jan. 2011.