A Fast Locking All-Digital Phase-Locked Loop via Feed-Forward Compensation Technique

Xin Chen, Jun Yang, *Member, IEEE*, and Long-Xing Shi, *Member, IEEE*

***Abstract*—A fast locking all-digital phase-locked loop (ADPLL) via feed-forward compensation technique is proposed in this paper. The implemented ADPLL has two operation modes which are frequency acquisition mode and phase acquisition mode. In frequency acquisition mode, the ADPLL achieves a fast frequency locking via the proposed feed-forward compensation algorithm. In phase acquisition mode, the ADPLL achieves a finer phase locking. To verify the proposed algorithm and architecture, the ADPLL design is implemented by SMIC 0.18-μm 1P6M CMOS technology. The core size of the ADPLL is 582.2 μm $*$ 343 μm. The frequency range of the ADPLL is from 4 to 416 MHz. The measurement results show that the ADPLL can achieve a frequency locking in two reference cycles when locking to 376 MHz. The corresponding power consumption is 11.394 mW.**

***Index Terms*—All-digital phase-locked loop (ADPLL), digitally controlled oscillator (DCO), feed-forward compensation technique, frequency divider.**

I. Introduction

PHASE-LOCKED LOOPS (PLLs) have been widely used as clock generators in system-on-chip (SoC) microprocessors. Traditionally, analog approaches are adopted to design PLLs. But it is difficult to integrate an analog PLL into a noisy SoC environment.

Therefore, it becomes popular to digitalize the implementation of the loop filter and oscillator circuits in PLLs [1], [2]. This kind of PLLs is named as all-digital phase-locked loop (ADPLL). Recent examples of ADPLL have been reported in [3]–[7].

Compared with the traditional analog PLLs, ADPLLs have several advantages. First, most of the signals in the ADPLLs are digital formats. Thereby, the ADPLLs have higher immunity to switching noise. Second, the ADPLLs can be implemented by electronic design automatic (EDA) tools. This can reduce design time greatly. Third, the analog filter is replaced with digital filter. The large area for the capacitors in the analog filter is saved. Therefore, the ADPLL is more readily scaled down in size when new fabrication processes are utilized. Fourth, the frequency of the digitally controlled oscillator (DCO) is tuned by digital codes. So ADPLLs are much easier to achieve fast frequency acquisitions.

Fast frequency acquisition is crucial for PLL used in an SoC processor. To reduce the power consumption, the SoC processor will enter the sleep mode when it is not in use. In the sleep mode, the processor will turn off several parts of the processor. This may include turning off the PLL. When the SoC processor exits the sleep mode and works again, the PLL should provide the processor with the correct clock as soon as possible. So, designing a fast locking PLL is very important for a SoC processor.

Usually, there are several frequency search algorithms used in ADPLLs. One of the typical methods is adjusting the PLL's loop bandwidth dynamically [8], [9]. This method is very common in fast locking charge pump PLLs [10]. As is well known, the locking time is directly proportional to the initial frequency difference between the reference clock and the divided clock, and inversely proportional to the loop bandwidth of the PLL. In [10], when the phase error between the reference clock and the divided clock is large, the PLL increases the loop bandwidth and achieves fast locking. Conversely, when the phase error is small, the PLL decreases the loop bandwidth and minimizes the output jitter. But this method will increase the complexity of the PLL circuits. The PLL must be stable over a wide range of the PLL's loop bandwidth, and must tolerate the errors in the prediction of the loop's parameters such as the oscillator gain. Furthermore, balancing the settling speed and stability of the PLL emerges as a new problem. Reference [11] obtains fast frequency acquisition by employing a digital phase/frequency detector (DPFD) and a variable loop gain scheme. The minimum locking time is obtained when the loop gain $K_I K_o/\omega_{\rm ref}$ equals one, where K_I is the proportional gain in the digital filter, K_o is the oscillator gain, and $\omega_{\rm REF}$ is the angular frequency of the reference clock. Since the oscillator gain is a parameter of process, voltage and temperature (PVT) variation, it is difficult to precisely predict the optimum K_I to achieve the minimum locking time. Seven-cycle locking time is achieved in [12] by a high-linearity ring oscillator. However, the frequency search algorithm in [12] depends on the structure of the oscillator heavily. It is hard to implement the frequency search algorithm with the other DCO structures. Besides, the frequency range of the oscillator implemented in [12] is hard to enlarge. Binary search algorithm (BSA) is widely used in ADPLLs [4], [13]. The BSA is implemented easily and can achieve a fast locking. The maximum locking time is proportional to $O(\log_2 n)$, where n is the number of discrete frequency point in the DCO. Thereby, a tradeoff among locking time, frequency range, and DCO gain must be made when the BSA is utilized in an ADPLL.

The feed-forward compensation technique predicts the desired code by the estimation value of several ADPLL parameters [14]. By sending the desired code directly to the oscillator, the frequency error between the reference clock and the divided clock is significantly reduced. However, due to the inaccuracies

Manuscript received May 21, 2009; revised September 02, 2009. First published February 17, 2010; current version published April 27, 2011.

The authors are with the National ASIC System Engineering Research Center, Southeast University, Nanjing 210096, China (e-mail: xinchen@seu.edu.cn).

Digital Object Identifier 10.1109/TVLSI.2009.2039971

resulting from the error of the estimation value, a feedback loop such as phase-locked loop is necessary to provide an adjustment in addition to the feed-forward compensation technique.

In this paper, a fast locking ADPLL via feed-forward compensation technique is described. The proposed ADPLL has two operation modes which are frequency acquisition mode and phase acquisition mode. In frequency acquisition mode, a feed-forward compensation structure is activated. It takes two reference cycles to estimate an ADPLL parameter which is called K_f here, and predicts the desired code with K_f. The predicted code is directly sent to the DCO. Then the ADPLL enters the phase acquisition mode, a PLL is activated to eliminate the remaining frequency error.

This paper is organized as follows. The feed-forward compensation algorithm is proposed in Section II. In Section III, the structure and operation of the proposed ADPLL is introduced. The detail structure of the proposed ADPLL is described in Section IV. The mathematical analysis of the proposed ADPLL is given in Section V. In Section VI, a design example is given. Finally, the improved design is described in Section VII, and conclusions are drawn in Section VIII.

II. Proposed Feed-Forward Compensation Algorithm

When the code W is assigned to tune the DCO, the DCO frequency is f. F is the corresponding value sensed by the modified frequency divider (MDIV). The operation of the MDIV will be explained in Section IV. Simply speaking, the MDIV counts by the rising edge of the DCO clock when the reference clock is low level. When the reference clock rises, the value of the counter in MDIV is saved as F. In the following, the subscripts on the symbols f and F accord with the subscript on the symbol W. So, when two codes W_1 and W_2 are assigned to tune the DCO successively, the corresponding frequencies of the DCO are f_1 and f_2. F_1 and F_2 are the corresponding values sensed by the MDIV.

The relation between the sensed value F and the frequency of the DCO f is given in (1)

$$F = \frac{f}{2f_{\text{ref}}} \tag{1}$$

where f_{ref} is the frequency of the reference clock.

Therefore, the frequency error between f_1 and f_2 can be represented by the difference between F_1 and F_2

$$F_1 - F_2 = \frac{(f_1 - f_2)}{2f_{\text{ref}}}. \tag{2}$$

In many DCO designs, f_1 and f_2 can be defined as (3)

$$f_1 = f_{\min} + K_o \bullet W_1, \quad f_2 = f_{\min} + K_o \bullet W_2 \tag{3}$$

where K_o is the DCO gain measured in megahertz, and $f_{\min}$ is the minimum frequency of the DCO.

Combining (2) with (3), the parameter K_f can be obtained by (4)

$$K_f = \frac{2f_{\text{ref}}}{K_o} = \frac{W_1 - W_2}{F_1 - F_2}. \tag{4}$$

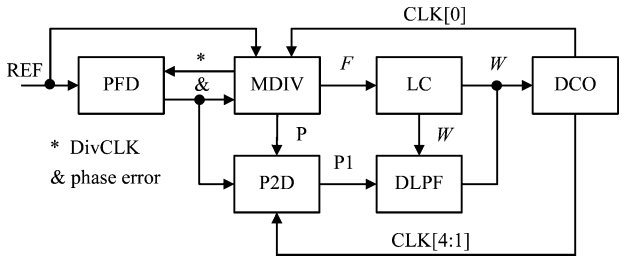


Fig. 1. Structure of the proposed ADPLL.

Then (4) can be rewritten as (5)

$$W_1 = W_2 + K_f \bullet (F_1 - F_2). \tag{5}$$

Because of the MDIV which will be described later, the stored value of F is $M/2$ when the ADPLL is frequency locking. M is the frequency divider ratio.

In (5), replacing F_1 with $M/2$, and using W_{locked} instead of W_1, then W_{locked} can be calculated by (6)

$$W_{\text{locked}} = W_2 + K_f \bullet \left(\frac{M}{2} - F_2\right). \tag{6}$$

From (6), it is seen that the ADPLL with the proposed algorithm can achieve a fast frequency locking when the values W_2 and $K_f \bullet (M/2 - F_2)$ are known. The code W_2 is generated by the ADPLL itself. The value of M is set to the ADPLL before the ADPLL works, and the value of F_2 is sensed by the MDIV, so these values can be obtained easily. The value which is the most difficult to obtain is K_f. From (4), it is seen that the parameter K_f is PVT dependent. So it is better to recalculate the value of K_f for every initialization of the ADPLL.

III. Structure and Operation of the Proposed ADPLL

The proposed ADPLL, shown in Fig. 1, includes a phase/frequency detector (PFD), an MDIV, a loop control (LC), a ring-type DCO, a first-order digital loop filter (DLPF), and a P2D. The stage of the DCO in the proposed ADPLL is five, so the DCO can provide five clocks whose phases are different. The five clocks are named as CLK[0], CLK[1], CLK[2], CLK[3], and CLK[4]. In Fig. 1, CLK[1], CLK[2], CLK[3], and CLK[4] are written as CLK[4 : 1] for short.

The operation of the frequency acquisition mode and phase acquisition mode is shown in Fig. 2. It includes four states (State $0 \sim 3$). The state status of the ADPLL is updated by the rising edge of signal Upd_state which is the delayed REF drawn in Fig. 4.

A. Frequency Acquisition Mode

In this mode, the feed-forward compensation structure which includes the MDIV, LC, and DCO is activated.

The MDIV is reused as a frequency detector in this mode. The output of the MDIV F sends to the LC.

The LC is a digital processing section. Depending on the value of F, it generates the next code and sends the code to the DCO and the DLPF.

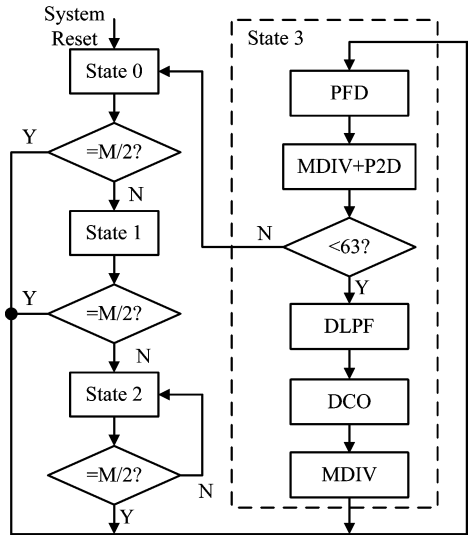


Fig. 2. Flow chart of the ADPLL.

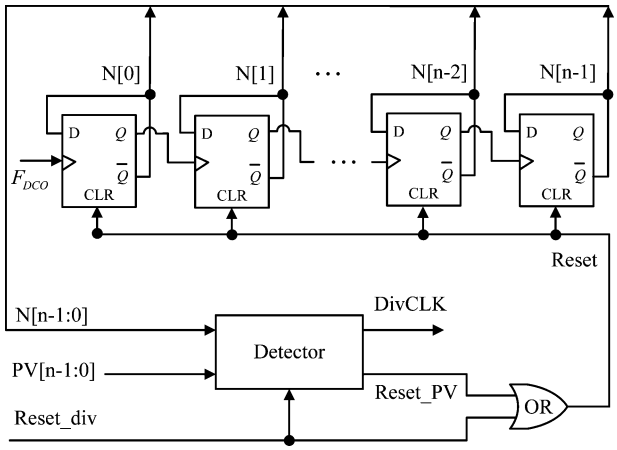


Fig. 3. Structure of the asynchronous DIV.

Based on the code, the DCO generates the DCO clock $\mathrm{CLK}[0]$ which is fed back to the MDIV.

The frequency acquisition mode consists of three states (State $0 \sim 2$). The frequency locking is achieved when the output of the MDIV F is equal to $M/2$.

In State 0, the LC generates the middle code W_1. The sensed value by the MDIV is F_1. If F_1 equals $M/2$, the ADPLL enters State 3, else the ADPLL enters State 1.

In State 1, if $F_1 > M/2$, it indicates that the frequency of the DCO is higher than the desired frequency and should be decreased. So the second code W_2 will be decreased compared with W_1. If $F_1 < M/2$, the second code W_2 will be increased. The corresponding output of the MDIV is F_2. If F_2 equals $M/2$, the ADPLL enters State 3, else the ADPLL enters State 2.

In State 2, the LC estimates the ADPLL parameter K_f with (4), and predicts the third code W_3 according to (6). If the corresponding output of the MDIV F_3 is not equal to $M/2$, the state will stay in State 2. The fourth code W_4 will be computed by the LC based on the third sensed frequency information F_3. During the rest frequency acquisition time in State 2, the value of K_f will be reduced to the half of the previous value only when the relation between the value of F and $M/2$ changes from large to small (or vice versa). This operation will continue until the ADPLL achieves a frequency locking. Furthermore, the minimum difference between the current code and the previous code is set to one. So the ADPLL will never enter an endless loop during State 2. When the frequency locking is achieved, the ADPLL enters phase acquisition mode which is named as State 3 in Fig. 2.

B. Phase Acquisition Mode

When the ADPLL is under phase acquisition mode, a PLL is activated to eliminate the remaining frequency error. The ADPLL achieves a phase locking when the code for the DCO oscillates between the neighboring codes.

The PFD senses the phase error between the divided clock (DivCLK) and reference clock (REF). The sensed phase error is converted to digital formats by the MDIV and the P2D. If the digitized phase error is larger than 63, the state will turn to State 0 at the rising edge of the signal Upd_state, or else the digital information is sent to the DLPF. Then the output of the DLPF tunes the DCO. Finally, the DCO clock is divided by the MDIV and is fed back to the PFD.

IV. Detail Structure of the ADPLL

A. Structure of the MDIV

Usually, the asynchronous frequency divider (DIV) is utilized to generate the DivCLK by the high frequency DCO clock. The structure of the asynchronous DIV is drawn in Fig. 3. When the signal Reset_div of the DIV is high level, the DivCLK is low level and the value of the counter N[n-1:0] keeps at 0. When the signal Reset_div is low level, the DIV counts by the rising edge of the DCO clock F_{DCO}. PV[n-1:0] is the modulus value of the DIV, and it is assigned to be one half of the frequency divider ratio. When N[n-1:0] equals PV[n-1:0], the DivCLK inverses its phase, and N[n-1:0] is reset to 0 immediately.

In the proposed ADPLL, the DIV is added by three modules which are SaveF module, Reset_syn module and T2D module. The DIV with the three modules are renamed as MDIV which is drawn in Fig. 4.

The SaveF module saves the value of the counter in the DIV when the REF rises. Because the DCO clock $\mathrm{CLK}[0]$ is not synchronous with the REF, the REF is retimed by the falling edge of the $\mathrm{CLK}[0]$. Then the value of counter N[n-1:0] will be stored in F[n-1:0] when the signal REF_d rises.

The function of the Reset_syn module is generating the reset signal Reset_div to control the operation of the DIV. When the signal Reset_div is low level, the DIV counts by the rising edge of $\mathrm{CLK}[0]$. When the signal Reset_div is high level, the DIV is reset for the next counting operation, and leaves the time for the LC to tune the DCO. The signal Reset_div turns to be high level under two conditions. First, the system reset signal in_Reset is high level. Second, the ADPLL is under frequency acquisition mode, the REF is high level, and the value of F[n-1:0] is saved by the SaveF module.

From Fig. 4, it is seen that the signal Reset_div is the OR-operation result of the signals in_Reset and Reset_alg. Signal Reset_alg is the AND-operation result of the signals REF, Upd,

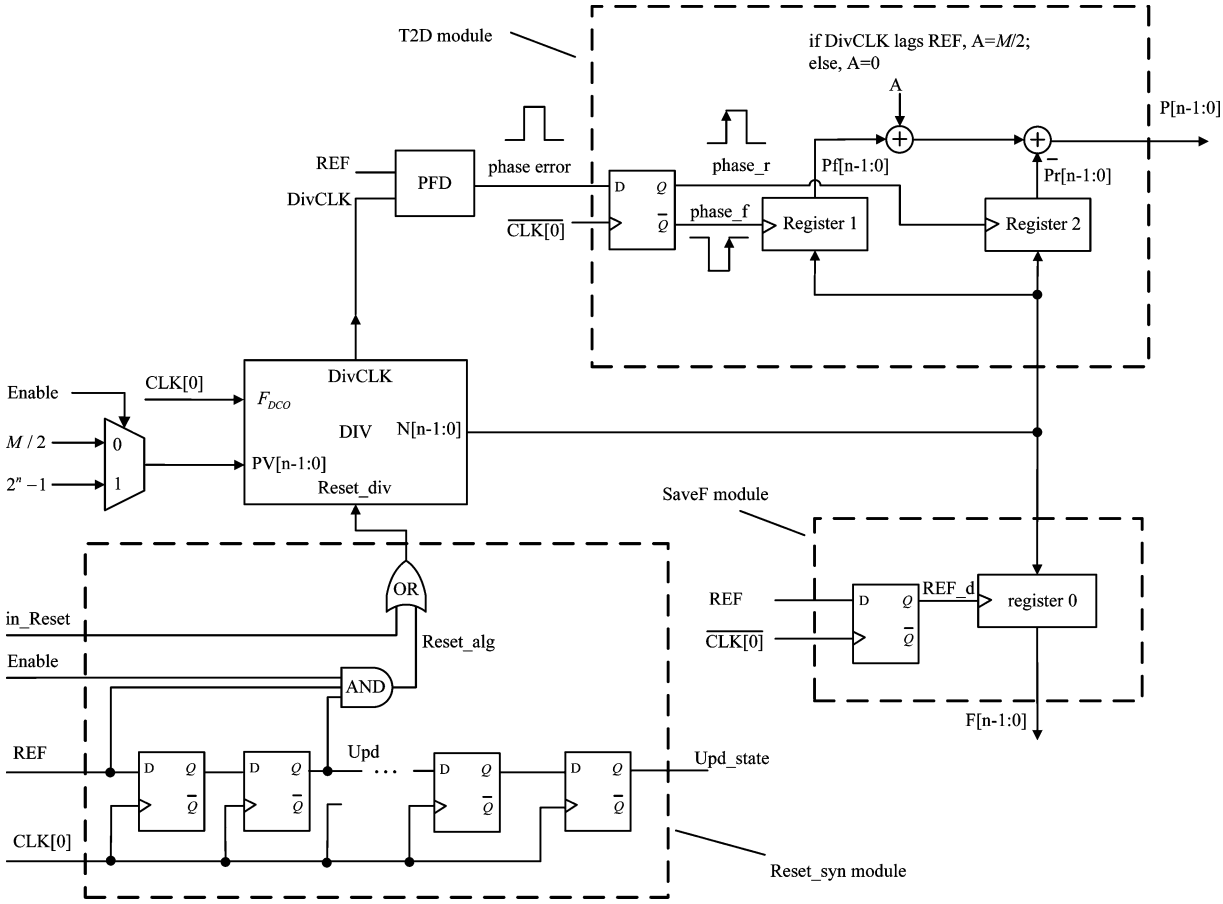


Fig. 4. Modified structure of the DIV.

and Enable. The signal Upd is the retimed REF which is delayed by two cycles of $\mathrm{CLK}[0]$. The signal Enable is the mode flag. It keeps high level during the frequency acquisition mode, and turns low by the first falling edge of the REF when the ADPLL enters phase acquisition mode. Therefore, the signal Reset_div turns high when the signal in_Reset is high level. If the signal in_Reset is low level and the signal Enable is high level, the signal Reset_div falls when the REF falls, and rises when the signal Upd rises. When the signal Enable is low level, the signals REF and Upd cannot affect the signal Reset_div.

In addition, the modulus value PV[n-1:0] in DIV is set to the largest value $2^n - 1$ when the signal Enable is high level. Therefore, the signal Reset_PV keeps low level and will not reset the counter in the DIV during the frequency acquisition mode. When the signal Enable falls, the modulus value PV[n-1:0] is set back to $M/2$, and the MDIV begins to divide the clock $\mathrm{CLK}[0]$. So the first sensed phase error is only dependent on the first period of the DivCLK and REF when the ADPLL enters the phase acquisition mode.

The T2D module converts the phase error to digital formats. The sensed phase error by the PFD is re-sampled on the falling edge of the clock $\mathrm{CLK}[0]$. Then the value of N[n-1:0] is stored in Pr[n-1:0] and Pf[n-1:0] at the rising edge of the signal phase_r and phase_f, respectively. Because the feed-forward compensation algorithm is adopted in this ADPLL, the sensed phase error is always smaller than the half cycle of the REF when the ADPLL is under phase acquisition mode. Therefore, when the DivCLK leads the REF, the digitized phase error will be Pf-Pr. If the DivCLK lags the REF, the digitized phase error will be equal to $M/2 + \mathrm{Pf} - \mathrm{Pr}$. The reason is that the value of the counter in the DIV is reset to zero when the signal DivCLK falls. So to obtain the correct value of the digitized phase error, the modulus value of the DIV which equals $M/2$ is added to Pf.

B. Structure of the LC

The LC is a digital processing section whose structure is shown in Fig. 5. It performs two operations when the ADPLL is initialized. First, the LC calculates K_f by (4). Second, it generates W_{locked} based on (6). By (4), the algorithm needs two reference cycles to obtain the parameter K_f. In the first reference cycle, the values of F_1 and W_1 are obtained. In the second reference cycle, the values of F_2 and W_2 are obtained. Then the value of K_f is calculated by a division operation. The dividend is $\Delta W = W_1 - W_2$, and the divisor is $\Delta F = F_1 - F_2$. With the obtained K_f, the LC predicts the code W by (6).

The structure of the divider in the LC is given in Fig. 6. The resolution of the result is 2^{-4}. In Fig. 6, ΔF, $2\Delta F \ldots 16\Delta F$ are subtracted from ΔW concurrently. Then the results $D(1)$, $D(2) \ldots D(16)$ are sent to a sign comparator. The sign comparator finds the value of i when the sign of $D(i)$ is different from the sign of $D(i+1)$. This operation is given in (7), where $\mathrm{Sig}(D)$ represents the sign of the D

$$\mathrm{Sig}(D(i))! = \mathrm{Sig}(D(i+1)). \tag{7}$$

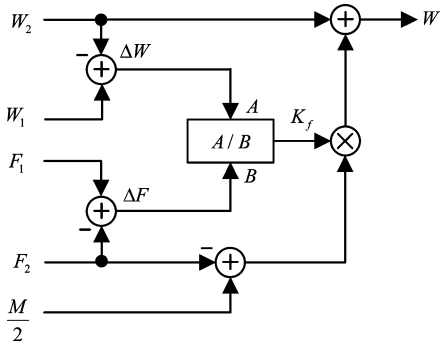


Fig. 5. Structure of the LC.

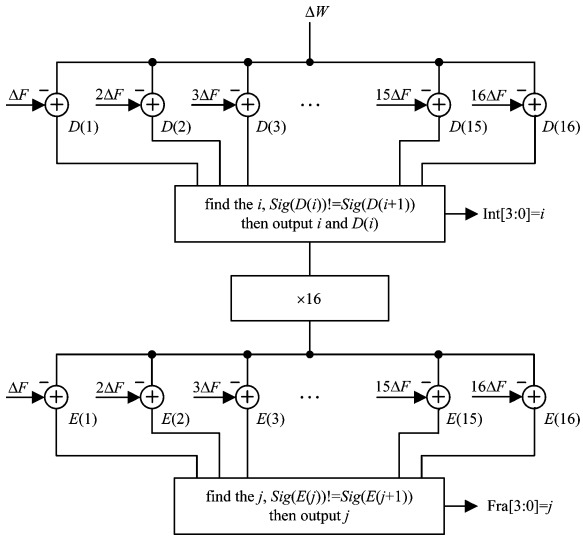


Fig. 6. Structure of the divider.

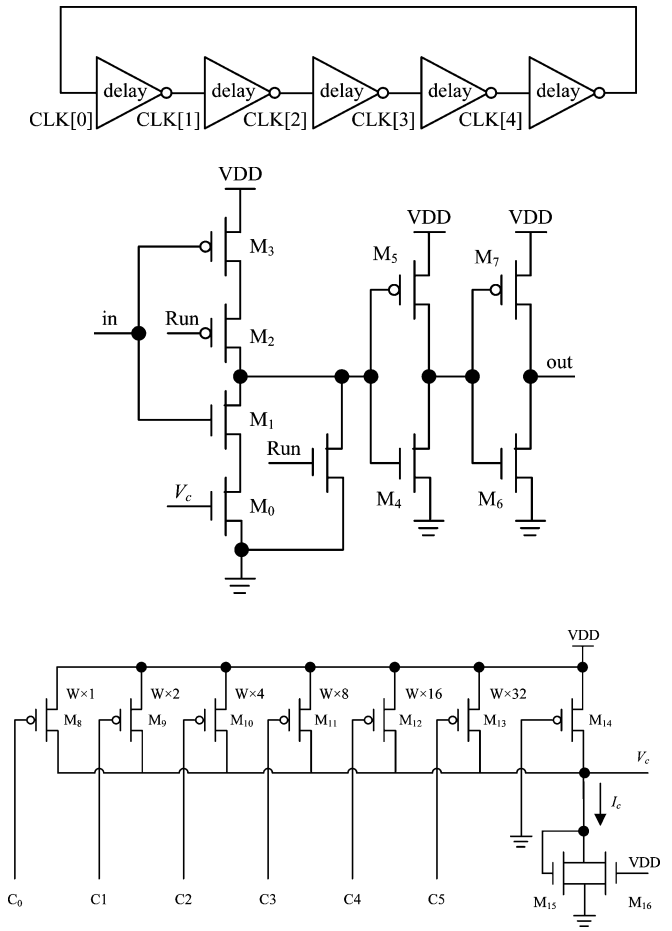


Fig. 7. (a) Structure of the DCO (b) The circuit of the delay cell. (c) Digital current controller.

Then $\Delta F, 2\Delta F \ldots 16\Delta F$ are subtracted from $16D(i)$, concurrently. The same structure of the sign comparator is utilized to find the value of j when (8) is valid

$$\mathrm{Sig}(E(j))! = \mathrm{Sig}(E(j+1)). \tag{8}$$

Finally, the result of the divider is $i + j/16$. For example, suppose ΔW equals 3.51 and ΔF equals 1. Then

$$D(3) = 0.51 > 0 \quad D(4) = -0.49 < 0.$$

So the value of i is 3. $D(3)$ is multiplied by 16. The obtained is 8.16. Then

$$E(8) = 0.16 > 0 \quad E(9) = -0.84 < 0.$$

So the value of j is 8. The error of the division operation is 0.01 which is less than 2^{-4}.

C. Structure of the DCO

The DCO is the combination of a digital-to-analog converter (DAC) and a voltage controlled oscillator (VCO). Based on the input code, the DAC converts the code to the voltage V_c. Then the voltage V_c controls the frequency of the VCO.

Seen from Fig. 7(a), the ring-type VCO can generate five clocks which are named as CLK[0]–CLK[4], respectively.

From Fig. 7(b), it is seen that the delay cell consists of a modified NOR cell and two inverter cells. In order to reduce power, the signal Run turns to be high level when the PLL is not in use. Then the output of the delay cell keeps low level. When the signal Run is low level, the voltage V_c controls the frequency of the VCO. The current through M_0 increases as the voltage V_c increases, so the delay time of the delay cell decreases and the frequency of the VCO increases (or vice versa).

Fig. 7(c) shows that the voltage V_c is generated by the digital current controller. The pMOS transistors $(M_8 \sim M_{13})$ are coded in a binary fashion, for example, the W/L ratio of M_9 is twice that of M_8, and so on. So, the range of the codes in the DCO is from 0 to 315. In order to obtain the minimum frequency of the DCO, M_{14} always keeps on. The sizes of M_{15} and M_{16} are the same. M_{15} and M_{16} can act as resistors. So the voltage V_c increases with the increase of the current I_c.

D. Structure of the PFD

The structure of the PFD is shown in Fig. 8. If the REF falls first, the signal UP is high level and the signal DOWN is low level. It indicates that the REF leads the DivCLK (or vice versa).

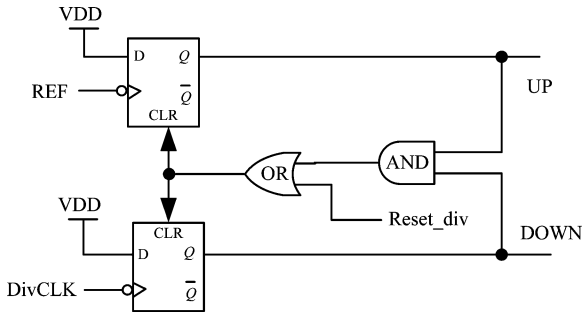


Fig. 8. Structure of the PFD.

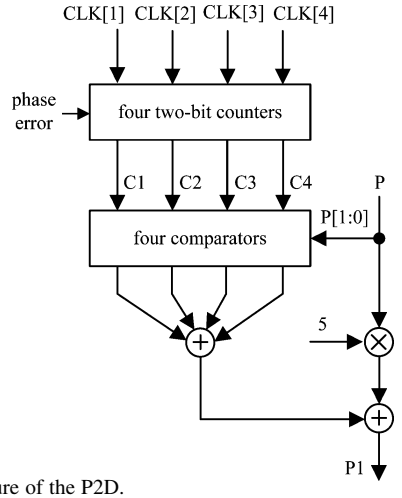


Fig. 9. Structure of the P2D.

If the signals UP and DOWN are both high levels, the two signals are both reset to low levels by a feedback reset signal. The sensed phase error by the PFD is the XOR-operation result of the signals UP and DOWN. Compared with the traditional PFD in [15], an OR gate is inserted into the reset path. When the signal Reset_div is high level, the D flip-flops in the PFD are reset. So during the frequency acquisition mode, the PFD does not sense the phase error.

E. Structure of the P2D

The resolution of the MDIV reused as the time-to-digital converter (TDC) is the period of the DCO clock. To reduce the quantization error of the MDIV reused as the TDC, the P2D module is added to count the phase error with the other four DCO clocks $\mathrm{CLK}[4:1]$.

The structure of the P2D is given in Fig. 9. It includes four two-bit counters, four comparators, one multiplier, and two adders.

The concept of the P2D module is proposed in [16]. If the value of the MDIV (for instance, P) is observed, the value of the other counters triggered by the other four clocks will be P or $P \pm 1$. Thus the other four counters can be replaced by four two-bit counters. A comparison between the four two-bit counters and the two lowest LSBs of the counter in the MDIV $P[1:0]$ yields the result 1, 0, or -1. Hence, the value sensed by the MDIV is multiplied by five and is added to the sum of the comparison results. Then the sum result P1 is the digitalized phase error. The quantization error is less than one fifth of the DCO clock period.

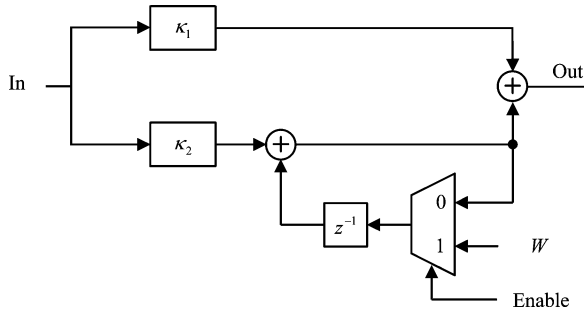


Fig. 10. Structure of the DLPF.

F. Structure of the DLPF

The structure of the DLPF is shown in Fig. 10. In Fig. 10, κ_1 and κ_2 are the DLPF parameters. When the signal Enable is high level, the code predicted by the LC is inserted into the integral path. Due to the structure of the PFD, the input of the DLPF keeps zero during the frequency acquisition mode. So, when the ADPLL enters the phase acquisition mode, the first code output by the DLPF is W_{locked}. Then the following codes are decided by the sensed phase error.

V. Mathematical Analysis of the Proposed ADPLL

A. Mathematical Analysis Under Frequency Acquisition Mode

Based on (6), the difference formula for the ADPLL under frequency acquisition mode is developed as

$$\begin{cases} f_{\mathrm{DCO}}(k) = f_{\min} + W(k) \bullet K_o \\ W(k) = W(k-1) + \left(\frac{M(k-1)}{2} - \frac{f_{\mathrm{DCO}}(k-1)}{2f_{\mathrm{ref}}} \right) \bullet K_f \end{cases} \tag{9}$$

where $f_{\mathrm{DCO}}(k)$ is the kth frequency of the DCO, $f_{\min}$ is the minimum frequency of the DCO, $W(k)$ is the kth code for the DCO, and $M(k)$ is the kth frequency divider ratio.

Combining (3) and (4) with (9), (10) is obtained

$$f_{\mathrm{DCO}}(k) = f_{\mathrm{ref}} \bullet M(k-1). \tag{10}$$

From (10), it is seen that the ADPLL during frequency acquisition mode is unconditional stable system in idea condition.

But in actual condition, the predicted code error will be introduced into the loop due to the following reasons. First, the DCO gain is hard to keep invariant during the whole frequency range of the DCO. Second, the resolution of the divider in the LC module is 2^{-4}. Third, the quantization error is added when the MDIV senses the frequency information F.

The feed-forward compensation algorithm is performed from the third code W_3 according to (6). From (6), it is seen that the predicted K_f is adversely affected by the above three errors, and F_2 is affected by the MDIV quantization error.

From (4), it is seen that the nonuniform DCO gain and the quantization errors of the MDIV and divider affect the precision of K_f separately. So the effects of the errors on the predicted code error are considered independently in the following.

Due to the nonuniform DCO gain, the predicted DCO gain K_o^p is $K_o(1-\delta)$, where δ is the DCO gain estimation error. The predicted code is given as

$$W_3^p = W_2 + K_f \bullet \left(\frac{M}{2} - F_2\right) = W_2 + \frac{f_{\text{locked}} - f_2}{K_o^p} = W_2 + \frac{f_{\text{locked}} - f_2}{K_o(1-\delta)}. \quad (11)$$

So the predicted code error due to the nonuniform DCO gain is given as

$$E_{\text{DCO}} = W_3^p - W_3 = \frac{f_{\text{locked}} - f_2}{K_o} \bullet \frac{\delta}{1-\delta}. \quad (12)$$

To minimize the negative influence of the predicted code error E_{DCO}, the absolute value of E_{DCO} should be less than one. Thus, the following inequalities should be met:

$$-\frac{K_o}{|f_{\text{locked}} - f_2| - K_o} < \delta < \frac{K_o}{|f_{\text{locked}} - f_2| + K_o}. \quad (13)$$

For simplicity, (13) is rewritten as (14)

$$-\frac{1}{|W_{\text{locked}} - W_2| - 1} < \delta < \frac{1}{|W_{\text{locked}} - W_2| + 1}. \quad (14)$$

From (14), it is seen that the tolerance to the nonuniform DCO gain increases while the difference between the second code W_2 and the desired code W_{locked} decreases.

When the quantization errors of the MDIV and the divider in the LC are considered, the second term in (6) is rewritten as

$$\begin{aligned} K_f \bullet \left(\frac{M}{2} - F_2\right) &= \left(\frac{M}{2} - F_2^s\right) \bullet \left[\frac{W_1 - W_2}{F_1^s - F_2^s} - q_{\text{div}} \bullet 2^{-m}\right] \\ &= \left(\frac{M}{2} - F_2 + q_2\right) \bullet \frac{W_1 - W_2}{F_1 - F_2 + q_2 - q_1} \\ &\quad - \left(\frac{M}{2} - F_2 + q_2\right) \bullet \frac{q_{\text{div}}}{2^m} \end{aligned} \quad (15)$$

where F_1^s and F_2^s are the first and the second sensed values by the MDIV, q_1 and q_2 are the corresponding quantization errors. Compared with F_1^s and F_2^s, F_1 and F_2 are the first and the second values without the quantization errors, and $q_{\text{div}} \bullet 2^{-m}$ is the quantization error introduced by the divisor in the LC. The values of the quantization errors q_1, q_2 and q_{div} are from zero to one. 2^{-m} is the resolution of the divider in the LC.

The first term in the result of (15) is rewritten as

$$\begin{aligned} &\left(\frac{M}{2} - F_2 + q_2\right) \bullet \frac{W_1 - W_2}{F_1 - F_2 + q_2 - q_1} \\ &= \left(\frac{M}{2} - F_2 + q_2\right) \bullet \frac{W_1 - W}{F_1 - F_2} \bullet \left(1 - \frac{q_2 - q_1}{F_1 - F_2 + q_2 - q_1}\right) \\ &= \left(\frac{M}{2} - F_2\right) \bullet K_f + K_f \bullet q_2 - K_f \bullet \frac{\left(\frac{M}{2} - F_2 + q_2\right) \bullet (q_2 - q_1)}{F_1 - F_2 + q_2 - q_1}. \end{aligned} \quad (16)$$

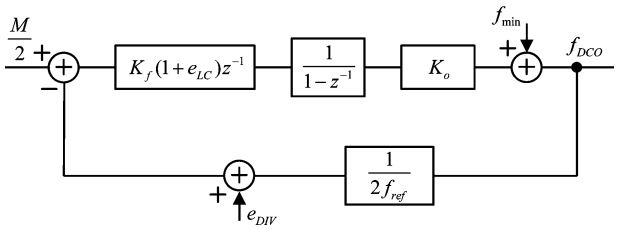


Fig. 11. ADPLL mathematical model during frequency acquisition mode.

The last two terms in the result of (16) can be concluded as the predicted error E_{MDIV} which is caused by the MDIV quantization error. The value of E_{MDIV} can be rewritten as follows:

$$\begin{aligned} E_{\text{MDIV}} &= K_f \bullet q_2 - K_f \bullet \frac{\left(\frac{M}{2} - F_2 + q_2\right) \bullet (q_2 - q_1)}{F_1 - F_2 + q_2 - q_1} \\ &< |K_f \bullet q_2| + \left| K_f \bullet \frac{(F_1 - F_2 + q_2) \bullet (q_2 - q_1)}{F_1 - F_2 + q_2 - q_1} \right| \\ &\approx K_f \bullet (|q_2| + |q_2 - q_1|). \end{aligned} \quad (17)$$

From (17), it is seen that the predicted error E_{MDIV} is proportional to K_f and the quantization errors. Hence, for low gain DCO, the predicted error E_{MDIV} will increase.

The second term in the result of (15) is the division operation error E_{MDIV}. To decrease the division operation error, the inequality (18) should be met

$$E_{\text{divider}} \approx \left(\frac{M}{2} - F_2\right) \bullet 2^{-m} < 1. \quad (18)$$

Thus, if the inequality (19) is met, the division operation error can be neglected

$$m > \log_2 \left(\frac{M}{2} - F_2\right). \quad (19)$$

B. Stability Analysis Under Frequency Acquisition Mode

The ADPLL mathematical model with errors under frequency acquisition mode is drawn in Fig. 11. In Fig. 11, the errors in estimating the parameter K_f and digitalizing the reference clock are modeled as the error e_{LC} and e_{DIV}.

Due to the above discussion, the error e_{LC} is caused by the nonuniform DCO gain, the MDIV quantization error and the division operation error. So e_{LC} is obtained as

$$e_{LC} = \frac{E_{\text{MDIV}} - E_{\text{DCO}} - E_{\text{divider}}}{K_f \left(\frac{M}{2} - F_2\right)}. \quad (20)$$

e_{DIV} is the quantization error and defined as

$$0 \le e_{\text{div}} < 1. \quad (21)$$

The transfer function of the ADPLL model with noises is given in (22)

$$f_{\text{DCO}} = \frac{2f_{\text{ref}} K_f (1 + e_{LC}) K_o z^{-1}}{2f_{\text{ref}} + (K_f K_o (1 + e_{LC}) - 2f_{\text{ref}}) z^{-1}} \frac{M}{2}$$

$$+\frac{2f_{\text{ref}}(1-z^{-1})}{2f_{\text{ref}}+(K_fK_o(1+e_{LC})-2f_{\text{ref}})z^{-1}}f_{\min}$$
$$-\frac{2f_{\text{ref}}K_f(1+e_{LC})K_oz^{-1}}{2f_{\text{ref}}+(K_fK_o(1+e_{LC})-2f_{\text{ref}})z^{-1}}e_{\text{DIV}}. \quad (22)$$

The unique pole is given in (23)

$$z=\frac{2f_{\text{ref}}-K_fK_o(1+e_{LC})}{2f_{\text{ref}}}. \quad (23)$$

The range of e_{LC} is given in (24) when the ADPLL is stable during the frequency acquisition mode

$$-1<e_{LC}<\frac{4f_{\text{ref}}}{K_fK_o}-1. \quad (24)$$

Combining (4) with (24), the range of e_{LC} is given as (25)

$$-1<e_{LC}<1. \quad (25)$$

After defining the predicted loop parameter $K_f'=K_f(1+e_{LC})$, (25) can be rewritten as (26).

$$0<\frac{K_f'}{K_f}<2. \quad (26)$$

If the value of K_f' is reduced by half, the tolerable range of the predicted K_f' is enlarged which is given in (27)

$$0<\frac{K_f'}{K_f}<4. \quad (27)$$

From (22), it is seen that the quantization error e_{DIV} adversely influences the locking performance of the ADPLL, but it does not affect the stability of the ADPLL.

However, the error e_{LC} in estimating the parameter K_f' plays an important role on the stability of the ADPLL during the frequency acquisition mode. To reduce the predicted error e_{LC} and decrease the effect of the predicted error e_{LC} on the stability of the ADPLL as much as possible, three methods are adopted in the ADPLL which are introduced in the following. First, to eliminate the influence of PVT variation on the value of K_f', the value of K_f' is calculated in the actual condition for every initialization of the ADPLL. Second, to decrease the variation of the DCO gain during the whole frequency range, the value of the second code depends on the first sensed F_1. So the desired code is not too far away from the first code W_1 and the second code W_2. Third, the value of K_f will be reduced to the half of the previous value when the sign of $M/2-F$ changes. Compared (26) with (27), the tolerable range of the predicted K_f' is doubled.

C. Stability Analysis Under Phase Acquisition Mode

The z-domain model of the ADPLL during phase acquisition mode is given in Fig. 12.

The z-domain model of the PFD is given by

$$\text{PFD}(z)=\frac{T_{\text{ref}}}{2\pi}z^{-1} \quad (28)$$

where T_{ref} is the period of the reference clock.

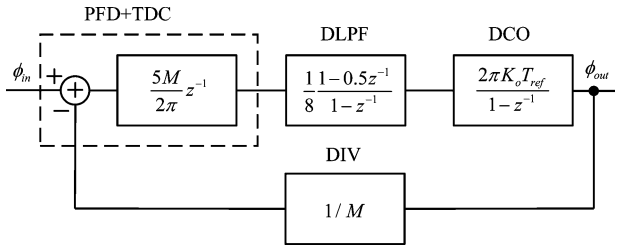


Fig. 12. z-domain model during phase acquisition mode.

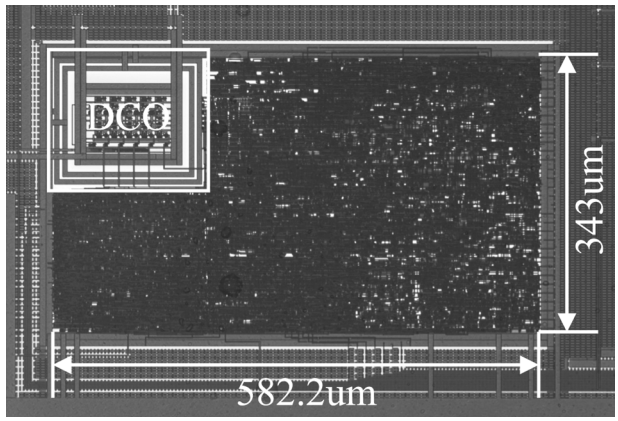


Fig. 13. Photomicrograph of the ADPLL core circuit.

The z-domain model of the TDC which includes the MDIV and P2D is given by (29)

$$\text{TDC}(z)=\frac{5}{T_{\text{DCO}}}=\frac{5M}{T_{\text{ref}}}. \quad (29)$$

So the z-domain model of the PFD and the TDC is

$$\text{PFD}(z)\bullet\text{TDC}(z)=\frac{5M}{2\pi}z^{-1}. \quad (30)$$

The structure of the DLPF is shown in Fig. 10, so the z-domain of the DLPF is

$$F(z)=\frac{\kappa_1+\kappa_2-\kappa_1z^{-1}}{1-z^{-1}} \quad (31)$$

where the values of κ_1 and κ_2 are both equal to 1/16 in the proposed ADPLL.

The difference equation for the DCO is given as follows:

$$\text{DCO}(k)=\text{DCO}(k-1)+2\pi K_oT_{\text{ref}}\bullet W(k) \quad (32)$$

where $\text{DCO}(k)$ is the kth phase of the DCO measured in radians.

Hence, the z-domain model of the DCO is given as follows:

$$\text{DCO}(z)=\frac{2\pi K_oT_{\text{ref}}}{1-z^{-1}}. \quad (33)$$

The z-domain model of the DIV can be modeled as $1/M$.

So the closed transfer function of the ADPLL is given by (34)

$$H(z)=\frac{5MK_o}{8f_{\text{ref}}}\frac{z-0.5}{z^2+(\kappa-2)z+1-0.5\kappa} \quad (34)$$

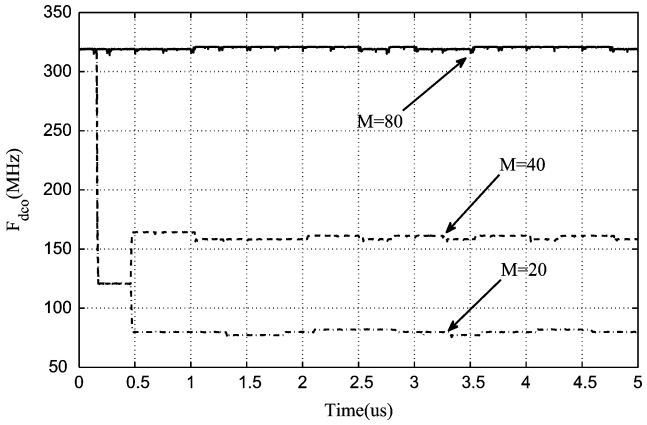


Fig. 14. Transient responses of the ADPLL.

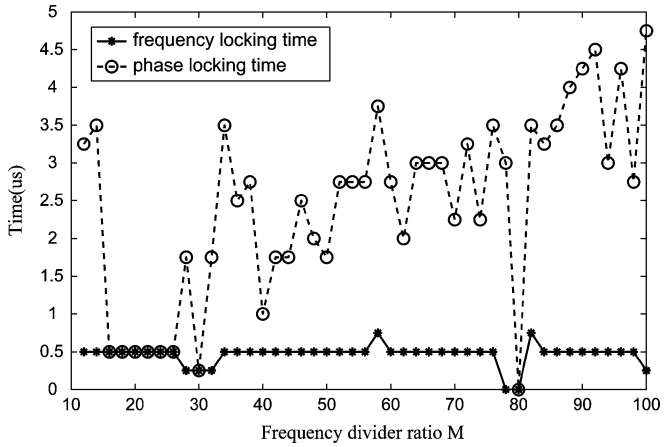


Fig. 15. Frequency locking time and phase locking time.

where $\kappa = 5K_o/8f_{\text{ref}}$.

The stability criteria obtained from (34) is given by (35)

$$0 < K_o < \frac{64}{15} f_{\text{ref}}. \tag{35}$$

Therefore, during the phase acquisition mode, the ADPLL is stable if the value of K_o is less than $64/15 f_{\text{ref}}$.

VI. Design Example

To verify the proposed algorithm and architecture, the ADPLL design is implemented by SMIC 0.18-μm 1P6M CMOS technology which is shown in Fig. 13.

The ring DCO shown in Fig. 7 is implemented by custom method in this ADPLL. The other components of the ADPLL are described by Verilog Hardware Description Language first. Then these codes are synthesized by Design Compiler. Finally, the layout of the ADPLL is realized by Astro. The core size of the ADPLL is 582.2 μm $*$ 343 μm.

After the layout of the ADPLL is implemented, several spice simulations are made. The frequency of the reference clock is 4 MHz for these simulations.

The transient responses of the ADPLL when the frequency divider ratio M equals 20, 40, and 80 are given in Fig. 14. From Fig. 14, it is seen that when M equals 20 and 40, the transient responses of the ADPLL before 0.5 μs are the same. The reason is that the first and the second codes for the two conditions are the same. After 0.5 μs, the desired codes are found, respectively. Then the respective frequencies of the DCO keep stable. When M equals 80, the ADPLL achieves a frequency locking when the first code is set. Then the ADPLL tunes to the phase acquisition mode to eliminate the remaining frequency error.

The frequency locking times and the phase locking times for multiple output frequencies are given in Fig. 15. The frequency locking time is from the time when the ADPLL is initialized to the time when the frequency locking is achieved. The phase locking time is from the time when the ADPLL is initialized to the time when the phase locking is achieved. So the phase locking time includes the frequency locking time. From Fig. 15, it is seen that most of the frequency locking times are 0.5 μs. When M is 58 or 82, the frequency locking times are 0.75 μs. When M is 78 or 80, the first code is the desired code, so the frequency locking times are 0 μs. When M is 28, 30, 32, or 100, the second code is the desired code, so the frequency locking times are 0.25 μs. In most conditions, the phase locking times are much larger than the frequency locking times. Most of the phase locking times are between 2 and 4 μs. The largest phase locking time is 4.75 μs. The phase locking time is directly proportional to the initial frequency difference between the reference clock and the divided clock, and inversely proportional to the loop bandwidth of the PLL. The initial frequency difference between the reference clock and the divided clock is given in Fig. 16. From Fig. 16, it is seen that the largest initial frequency difference is 0.31 MHz which is caused by the quantization error of the MDIV. So the phase locking time is still limited by the bandwidth of the DLPF.

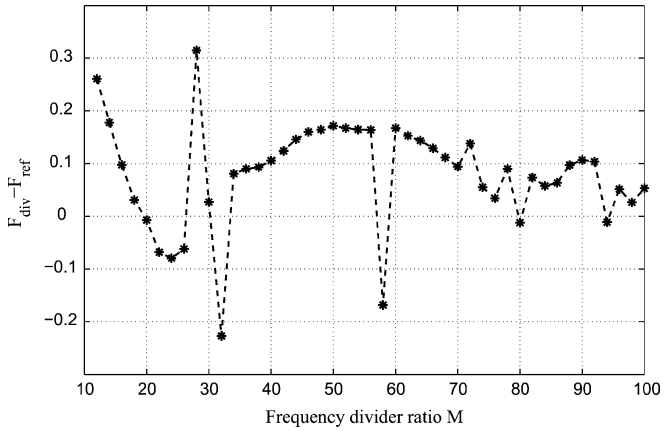


Fig. 16. Initial frequency difference between the reference clock and the divided clock.

Due to the speed limitation of the I/O pad, the output frequency of the DCO is divide-by-4. The measured frequency range is from 1 to 104 MHz by LeCory 204MXi. So, the frequency range of the ADPLL is from 4 to 416 MHz. An example of the frequency locking is shown in Fig. 17. The frequency of the reference clock is 4 MHz. The frequency divider ratio M is 94, and the desired frequency of the DCO is 376 MHz $(4 * 94)$. The measured frequency is the four-divided frequency of the DCO. From Fig. 17, it is seen that the measured frequency is 83.85 MHz in the first reference cycle. In the second reference cycle, the frequency of the four-divided DCO clock is 104.15

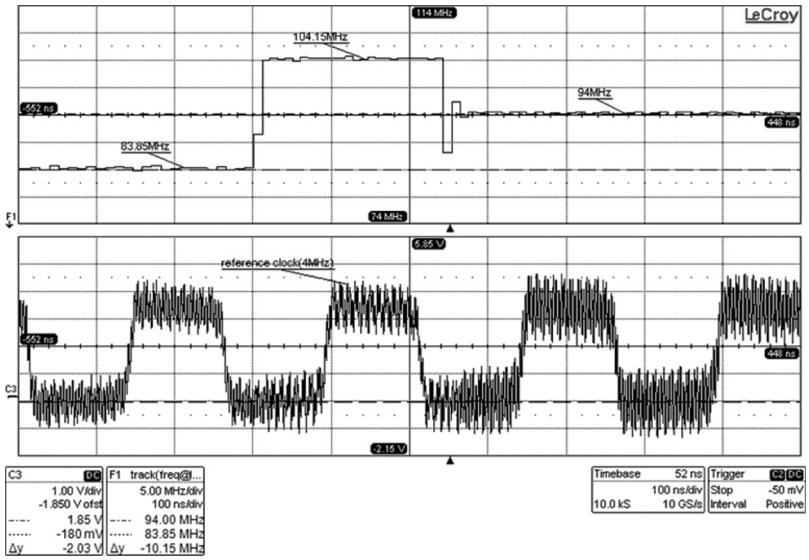


Fig. 17. Measured frequency acquisition when the ADPLL locks to 376 MHz.

TABLE I
PERFORMANCE COMPARISONS

Performance Parameter	This work	JSSC'06 [13]	JSSC'03 [4]	JSSC'01 [11]	JSSC' 03 [12]
Process	0.18μm CMOS	0.18μm CMOS	0.35μm CMOS	0.60μm CMOS	0.65μm CMOS
Area	0.2 mm^2	0.16 mm^2	0.71 mm^2	0.83 mm^2	1.17 mm^2
Power	11.394mW @ 376 MHz	15 mW @144MHz	100 mW @500MHz	105 mW @400MHz	not reported
Locking time	<3 cycles(frequency locking) <19 cycles(phase locking)	<75 cycles	<46 cycles	<16 cycles	<7 cycles
Max. Frequency	416MHz	378MHz	510MHz	934MHz	61.3MHz
Min. Frequency	4MHz	2.4MHz	45MHz	57.2MHz	0.0449MHz

MHz. In the third reference cycle, the frequency of the four-divided DCO clock is 94 MHz. Therefore, the ADPLL can achieve the frequency locking in 2 reference cycles when the ADPLL locks to 376 MHz. The corresponding measured current of the DCO is 5.28 mA and the current of the other components is 1.05 mA. So the whole power consumption is 11.394 mW.

Table I compares the results of the proposed ADPLL with four other PLLs. The area of the proposed ADPLL is larger than the ADPLL [13]. The frequency range of the proposed ADPLL is smaller than the ADPLLs [4], [11]. Although the frequency locking time of the proposed ADPLL is small, the maximum phase locking time is 19 cycles which is larger than the ADPLLs [11], [12]. However, the proposed ADPLL has the smallest power consumption.

VII. IMPROVED DESIGN

Although the first prototype of the PLL is fully functional, some major improvements are found to be possible to enhance the performance of the proposed ADPLL. This part describes the enhanced components.

A. Structure of the Enhanced MDIV

According to (17), the predicted error E_{MDIV} is proportional to K_f and the quantization error of the MDIV.

Because the quantization error of the MDIV reused as TDC can be reduced with the help of the P2D, the quantization error of the MDIV reused as frequency detector can also be reduced according to the same concept.

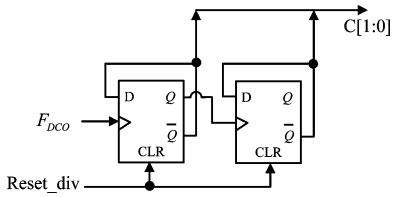


Fig. 18. Structure of the two-bit counter.

The structure of the two-bit counter is shown in Fig. 18. In Fig. 18, the signal Reset_div is the signal in the MDIV which is drawn in Fig. 4. The signal F_{DCO} is the DCO clock. The output of the two-bit counter is $C[1:0]$.

The structure of the enhanced MDIV is drawn in Fig. 19 which is similar to the structure in Fig. 9. But the enhanced MDIV in Fig. 19 has two modes, which are frequency-to-digital (F2D) mode and phase-to-digital (P2D) mode. The mode state is decided by the signal Enable. When the signal Enable is high level, the enhanced MDIV is under F2D mode. If the signal Reset_div is low level, the four two-bit counters counts by the rising edge of $\mathrm{CLK}[4:1]$, respectively. The signal $\mathrm{CLK}[0]$ is fed into the MDIV. The corresponding results of the four counters are C1, C2, C3, and C4. The value of R equals F. The four comparators compare C1, C2, C3, and C4 with $\mathrm{R}[1:0]$, and yield the results 1, 0, or -1, respectively. Hence, the result from the MDIV is multiplied by five before adding to the sum of the comparison results. Finally, the output of the enhanced MDIV is the frequency information F1. If the signal Enable is low level,

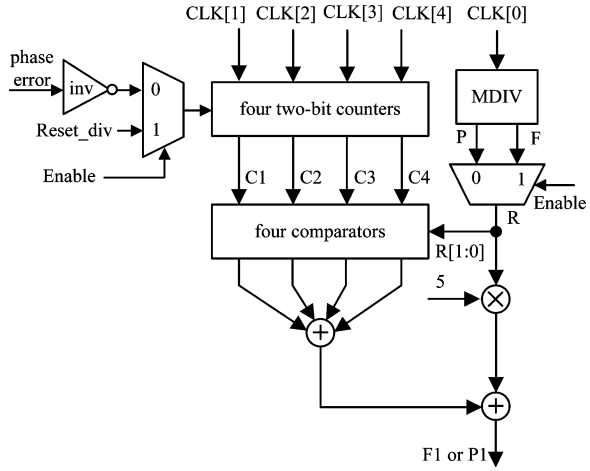


Fig. 19. Structure of the enhanced MDIV.

the enhanced MDIV is under P2D mode. The four counters counts when the signal phase error is high level. The value of R equals P. So the output of the enhanced MDIV is the digitalized phase error P1. With the help of the four two-bit counters, the quantization error of the MDIV reused as frequency detector is also reduced by five times.

B. Structure of the LC

The implemented ADPLL takes two cycles to calculate the parameter K_f, then predicts the desired code with K_f. The second code W_2 is predetermined and the first frequency information F_1 is not fully utilized. Moreover, according to (14), the influence of the nonuniform DCO gain on the predicted error decreases as the difference between the second code W_2 and the desired code W_{locked} decreases.

Therefore, a value of K_f based on the simulation result or sensed by the last time can be set to the ADPLL when it is initialized. When the frequency information F_1 is obtained, the second code is calculated according to F_1 and the saved K_f. Then the parameter K_f is calculated according to (4) and the third code is predicted by (6).

C. Structure of the DCO

The DCO gain can be decreased by two methods. First, decrease the W/L ratio of the PMOSFETs in the DAC to decrease the gain of the DCO. Second, employ a high-speed ÄÓ modulator to dither the finest code [1].

D. Structure of the DLPF

From Fig. 15, it is seen that the phase locking time is much larger than the frequency locking time. To decrease the phase locking time, the parameter of the DLPF should be designed to vary with the sensed phase error [9].

VIII. Conclusion

A fast locking ADPLL is proposed in this paper. To reduce the locking time, a feed-forward compensation algorithm is proposed in this ADPLL. Furthermore, the frequency divider is fully reused. The predicted error due to the estimation errors and the stabilities of the proposed ADPLL during the frequency acquisition mode and phase acquisition mode are analyzed in detail. Finally, the ADPLL design is implemented by SMIC 0.18-μm 1P6M CMOS technology. The area of the implemented ADPLL is 582.2 μm $*$ 343 μm and the frequency range of the ADPLL is 4–416 MHz. The measured results show that the ADPLL can complete frequency locking in 2 reference cycles, when locking to 376 MHz. The corresponding power consumption is 11.394 mW. However, based on the simulation results, the maximum frequency locking time is three cycles. Furthermore, most of the phase locking times are much larger than the frequency locking times. The largest phase locking time is 19 reference cycles. So according to Section VII, some major improvements are found to be possible to enhance the performance of the proposed ADPLL.

Acknowledgment

The authors would like to thank the associate editor and reviewers for their invaluable suggestions.

References

[1] R. B. Staszewski, D. Leipold, K. Muhammad, and P. T. Balsara, "Digitally controlled oscillator (DCO)-based architecture for RF frequency synthesis in a deep-submicrometer CMOS process," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 50, no. 11, pp. 815–828, Nov. 2003.

[2] P.-L. Chen, C.-C. Chung, and C.-Y. Lee, "A portable digitally controlled oscillator using novel varactors," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 5, pp. 233–237, May 2005.

[3] P.-Y. Wang, H.-H. Chang, and J.-H. C. Zhan, "Fractional spur reduction technique for RF TDC-based all digital PLLs," in *Proc. 34th Eur. Solid-State Circuits Conf.*, 2008, pp. 422–425.

[4] 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.

[5] R. B. Staszewski, J. L. Wallberg, S. Rezeq, C.-M. Hung, O. E. Eliezer, S. K. Vemulapalli, C. Fernando, K. Maggio, R. Staszewski, N. Barton, M.-C. Lee, P. Cruise, M. Entezari, K. Muhammad, and D. Leipold, "All-digital PLL and transmitter for mobile phones," *IEEE J. Solid-State Circuits*, vol. 40, no. 12, pp. 2469–2482, Dec. 2005.

[6] R. B. Staszewski and P. T. Balsara, "Phase-domain all-digital phase-locked loop," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 3, p. 159, Mar. 2005.

[7] V. Kratyuk and P. K. Hanumolu, "A design procedure for all-digital phase-locked loops based on a charge-pump phase-locked loop analogy," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 3, pp. 247–251, Mar. 2007.

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

[9] H. Lee, A. Bansal, Y. Frans, J. Zerbe, S. Sidiropoulos, and M. Horowitz, "Improving CDR performance via estimation," in *IEEE ISSCC Dig. Tech. Papers*, 2006, pp. 1296–1303.

[10] J. Lee and B. Kim, "A low-noise fast-lock phase-locked loop with adaptive bandwidth control," *IEEE J. Solid-State Circuits*, vol. 35, no. 8, pp. 1137–1145, Aug. 2000.

[11] I. Hwang, S. Lee, and S. Kim, "A digitally controlled phase-locked loop with a digital phase-frequency detector for fast acquisition," *IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1574–1581, Oct. 2001.

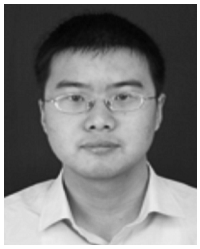
[12] T. Watanabe and S. Yamauchi, "An all-digital PLL for frequency multiplication by 4 to 1022 with seven-cycle lock time," *IEEE J. Solid-State Circuits*, vol. 38, no. 2, pp. 198–204, Feb. 2003.

[13] P.-L. Chen, C.-C. Chung, J.-N. Yang, and C.-Y. Lee, "A clock generator with cascaded dynamic frequency counting loops for wide multiplication range applications," *IEEE J. Solid-State Circuits*, vol. 41, no. 6, pp. 1275–1284, Jun. 2006.

[14] B. Zhang and P. Allen, "Feed-forward compensated high switching speed digital phase-locked loop frequency synthesizer," in *Proc. ISCAS*, Jun. 1999, pp. 371–374.

[15] R. Best, *Phase-Locked Loops—Design, Simulation, and Applications*, 5th ed. New York: McGraw-Hill, 2003.

[16] T. Olsson and P. Nilsson, "A digitally controlled PLL for SoC applications," *IEEE J. Solid-State Circuits*, vol. 39, no. 5, pp. 751–759, May 2004.



Xin Chen received the B.S. degree in electrical engineering from Southeast University, China, in 2001. He is currently pursuing the Ph.D. degree in microelectronics and solid electronics from the School of Electronic Science and Engineering, Southeast University, China.

His research interests mainly include mixed-signal circuits design, and digitally controlled phase-locked loop.

Jun Yang (M'08) received the B.S. degree in electronics engineering from Southeast University, China, in 1999, and the M.S. and Ph.D. degrees in microelectronics and solid electronics from Southeast University, China, in 2001 and 2004, respectively.

In 2004, he joined the faculty of the School of Electronic Science and Engineering, Southeast University, where he is currently a Professor. His research interests mainly include VLSI algorithms and architectures for high-speed networking, mixed-signal circuits design, system-on-chip design technology, and multimedia signal processing.

Long-Xing Shi (M'08) received the B.S. degree in radio engineering, the M.S. degree in electrical engineering, and the Ph.D. degree in microelectronics and solid electronics from Southeast University, China, in 1984, 1987, and 1992, respectively.

In 1987, he joined the faculty of the School of Electronic Science and Engineering, Southeast University, where he is currently a Professor. His research interests mainly include design methodology on mixed digital & analog and high-low voltage compatible IC, and design methodology on system-on-chip.