A Compact, Low-Power Low-Jitter Digital PLL

Amr M. Fahim

Qualcomm Inc., 5775 Morehouse Drive, San Diego CA 92121, USA
Email: afahim@qualcomm.com

Abstract:

This paper describes a compact, low-power and low-jitter digital PLL (DPLL). Digitizing the loop filter presents numerous challenges and advantages. In the proposed scheme, a wide-band DPLL is described that achieves low-jitter and low-power consumption. Low-jitter is achieved by using an all-digital adaptive bandwidth control scheme that can track the noise injected into the DPLL and optimise the loop bandwidth accordingly. A prototype DPLL employing has been implemented in a 0.25um CMOS technology. Measurement results show that jitter 130ps jitter is achieved at 144MHz. The area and power consumption outperform the traditional charge-pump based PLL by a factor of 3.5x and 45%, respectively.

1. Introduction

With ever increasing performance requirements of microprocessors, more stringent requirements have been placed on clock generation networks. At heart of the clock generation networks is the phase-locked loop (PLL). For system-on-a-chip (SOC) portable systems requirements on PLLs include low-area, low-power consumption, wide and flexible operating frequency range, and lowest possible jitter performance.

In this paper, a compact, low-power, low-jitter digital PLL (DPLL) is presented. In contrast to most other DPLL implementations, this DPLL is capable of having similar, if not better, performance than the analog charge-pump based PLLs. This has been made possible through a combination of several techniques. Low-power has been made possible by the elimination of extra overhead usually associated with analog PLLs, such as current for charge pumps, current generators, voltage-to-current converters, etc. The overall power savings result in 45%. Low-area has also been made possible by the elimination of the large analog-based loop filter and replacing it with a compact digital filter.

In Section 2, a survey of recent DPLL implementations and the proposed DPLL architecture are described. In Section 3, a system level analysis of loop dynamics and parameter optimisation are given. In Section 4, circuit techniques used to implement key blocks is given. In Section 5, the measured results and a comparison with other recent PLL implementations are given. Finally, in Section 6, overall results are summarized and conclusions are drawn.

2. DPLL Architecture

A diagram of a charge-pump based PLL is shown in Fig. 1. A PLL consists of five main blocks: a phase-frequency detector (PFD), a charge pump, a loop filter (usually a 2nd order RC filter), a voltage controlled oscillator (VCO), and a frequency divider of ratio N.

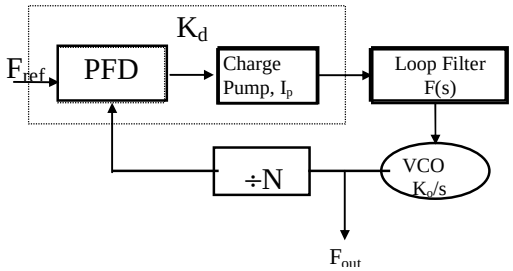

Figure 1: Conventional Charge Pump PLL

One of the most challenging blocks to integrate is the loop filter. For clock generation PLLs, the loop filter can consume as much as 50% of the total area. The capacitors are usually implemented as MOSFET gate capacitors. For large system-on-a-chip (SOC) applications, the PLL shares the same substrate as a large digital block. Much of the digital noise can couple through the substrate. Special technologies and techniques such as deep trench isolation, high resistivity substrates [3], and large guard rings are usually required for good isolation. As device sizes shrink, the oxide thickness also shrinks. This is an important factor enabling supply voltage scaling. This is problematic, however, for PLL designs, since the gate leakage increases exponentially with oxide thickness reduction [3]. Gate leakage, in this case, would cause large reference spurs, which increases jitter and ultimately can result in loss of lock.

In this study, an alternative method for PLL design, which is amenable to SOC mixed-signal design is explored. A PLL with a digital loop filter is explored. The main advantage of the DPLL is that it avoids all the issues listed above and its performance actually improves with technology.

The main challenge in DPLL design is obtaining a wide closed loop bandwidth while maintaining high frequency resolution. Frequency resolution limitation comes from the fact that the PFD can only quantize the

phase error by a high frequency clock, such as the VCO. In other words, if there are more VCO cycles per reference period, high frequency resolution can be obtained. However, it also means that the closed loop bandwidth of the system is also smaller. In one previous DPLL implementation, this was done by effectively creating a frequency locked loop [2]. The VCO phase was reset with every rising edge of the reference clock signal. Although this helped to recover significant frequency resolution, the DPLL still suffered from severe coarse granularity of frequency in the PFD, and hence its jitter performance was limited.

In [3], another solution the PFD granularity issue was presented. In this scheme, the loop filter is updated once every *m* cycles, where *m* is an integer multiple of the reference frequency. For higher accuracy, more PFD measurements are taken before updating the loop filter. This introduces a large loop delay which degrades DPLL stability and severely limits the achievable closed loop bandwidth.

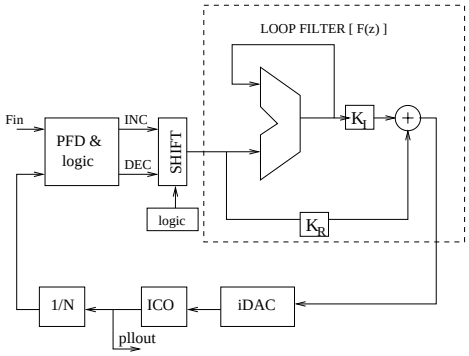

Figure 2: DPLL Architecture

In this study, an alternate method is used to obtain high frequency accuracy while maintaining a high loop bandwidth is presented. Fig. 2 shows a block diagram of the digital PLL (DPLL). The PFD is reduced to a one-bit comparator, which simply outputs the sign of the error. Initially, the error is assumed to be equal to one-half the size of the MSB. Once the sign of the error is changed, the amplitude of the error is also reduced by half. This allows for a binary tuning of the frequency, which is very fast. This is continued until the LSB is reached. At this point, the error is always assumed to be only one LSB. This is valid, since, once the PLL is locked, the error produced by the PFD can be assumed to be very small. A state machine illustrating this method is shown in Fig. 3. The "shift right" statements refer to reducing the size of the error by one-half.

The loop filter is updated only when the PFD outputs are updated. When the system is locked, this should be the same rate as the reference frequency. Once the loop filter is updated, the new value is sent to the current digital-to-analog converter (iDAC). The iDAC injects current into the current controlled oscillator (ICO), as shown in Fig. 2. The iDAC consists of an array of current sources that are digitally controlled to be on or off.

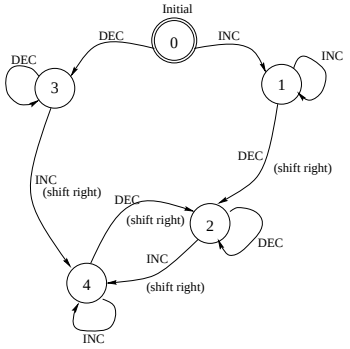

Figure 3: State Machine of Locking

Note that loop filter parameters K_R and K_I can be used to control the DPLL closed loop bandwidth. For PLLs using ring oscillators, much of the jitter would be due to ICO noise. Generally speaking, the wide loop bandwidth is desired to minimize ICO noise contribution to jitter. However, if the loop bandwidth is too wide, jitter due to the input signal may start to dominate.

In this implementation, an adaptive loop bandwidth mechanism is used to minimize jitter. During locked condition, the PFD should not produce several consecutive INC or DEC pulses. Jitter is estimated by counting the number of consecutive INC and DEC pulses. The values of K_R and K_I are adjusted to optimise the damping factor and bandwidth, respectively.

3. DPLL Loop Analysis

In this section, a detailed loop analysis is given.

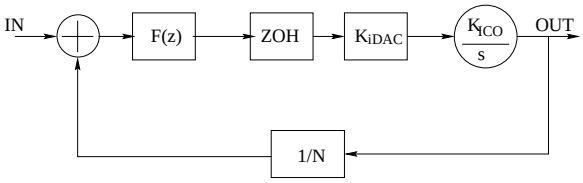

Figure 4: Linear discrete-time model of DPLL

When locked, the DPLL behaves as a linear discrete-time system. Fig. 4 shows a linear model of the DPLL. The iDAC is modelled as a zero-order hold filter (ZOH). F(z) is the loop filter transfer function, which is given as

$$F(z) = K_R + \frac{K_I}{1 - z^{-1}} \tag{1}$$

where K_R and K_I are the forward path and accumulator path coefficients, respectively. The closed loop expression of the DPLL transfer function is given as

$$H(z) = \frac{K \cdot (K_R + K_I) \cdot \left[Z - \frac{K_R}{K_R + K_I} \right]}{Z^2 - \left(2 - \frac{K \cdot (K_R + K_I)}{N} \right) \cdot Z + \left(1 - \frac{K}{N} K_R \right)} \tag{2}$$

where K is the forward gain coefficient given as K_{iDAC} * K_{ico}, the iDAC and ICO gain coefficients, respectively, and N is the feedback division ratio.

First, the simple case of K_I=K_R=1 is considered. The root locus of the system given by equation (2) is shown in Fig. 5. As shown, the system is unconditionally stable with a zero at 0.5 on the z-plane. As the open loop dc gain increases, the poles migrate towards the origin. At this point, the forward gain is equal to 2.

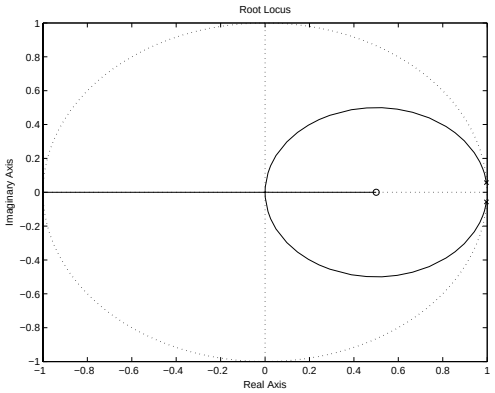

Figure 5: Root locus of DPLL with K_I=K_R=1

As the value of K_R is increased, the size of the circular radius of the pole movement shrinks, as shown in Fig. 6. It is evident from the figure, the K_R can be effectively used to control the damping factor, and hence, the locking characteristics of the DPLL. Therefore, K_R can be used to control the damping factor of the DPLL. K_R has been found to have small impact on the closed loop bandwidth of the DPLL.

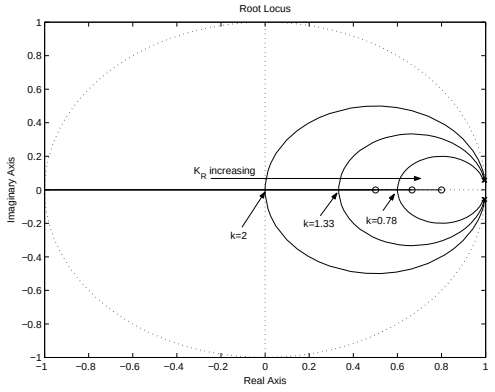

Figure 6: Effect of K_R on loop dynamics

K_I affects both the loop bandwidth and damping factor. As K_I is increased, the circular radius of the pole movement is also increased, as shown in Fig. 7. As K_I is increased the system will be more oscillatory and the positions of the pole would move faster along the z-plane. This means that the closed loop bandwidth of the DPLL is a strong function of K_I.

4. Circuit Implementation

In this section, key circuit blocks are given. The digital logic and high-speed feedback divider were implemented using a standard cell library. High-speed addition logic, such as carry-look ahead adders (CLA) [3] was used to implement the loop filters.

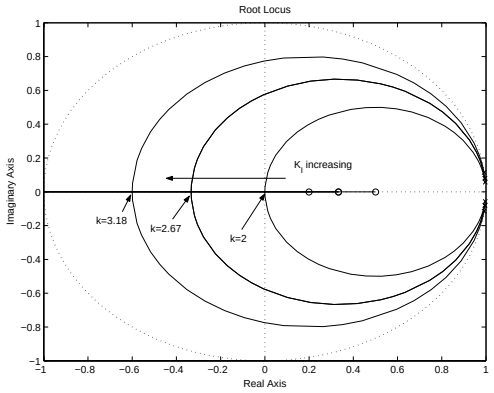

Figure 7: Effect of K_I on loop dynamics

The 10-bit iDAC was partitioned into two segments, as shown in Fig. 8. This was done to reduce power consumption. When the PLL is locked, only a few bits of the LSB will toggle to track small jitter variations over time, while the MSB bits will be fixed. For this reason, the MSB bits have been made single-ended, while the LSBs were made current steering. Single-ended current sources have the advantage of dissipating zero current while off. Current steering current sources have the advantage of having low switching ripple (and hence less jitter). Cascode PMOS current sources were used in both the MSB and LSB portions of the iDAC to enhance DAC linearity and to provide good immunity to low frequency supply noise. The MSB and LSB DAC arrays were thermally decoded and interdigitated to reduce gradient linearity errors [4].

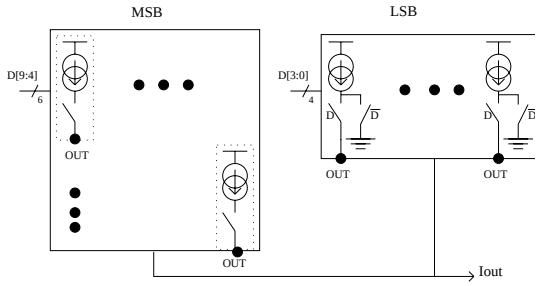

Figure 8: iDAC Partitioning for low-power

A three stage ring oscillator was used for the ICO. The delay stages consisted of differential pairs with PMOS symmetric loads [5]. This configuration was chosen for its good immunity to supply noise and its high linearity range. The PMOS current sources from the iDAC are dumped directly into the ICO. No additional filtering is required between the iDAC and the ICO.

5. Experimental Results

The DPLL has been fabricated in a 0.25um CMOS technology. The frequency spectrum of the DPLL is shown in Fig. 9. As shown, the close-in phase noise is less than –87dBc/Hz and the closed-loop bandwidth is

around 2MHz with a reference frequency of 19.2MHz.

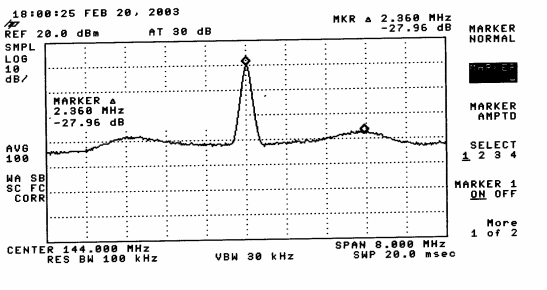

Figure 9: Frequency spectrum of DPLL

Fig. 10 shows the jitter plot of the DPLL. As shown, the DPLL exhibits only 130psec cycle jitter under noisy conditions (microprocessor on). Cycle jitter is defined as the average period minus the minimum period. This is the most relevant definition of jitter for microprocessor applications since it defines how much margin is subtracted from the clock period due to jitter. The DPLL performance is summarized in Table 1.

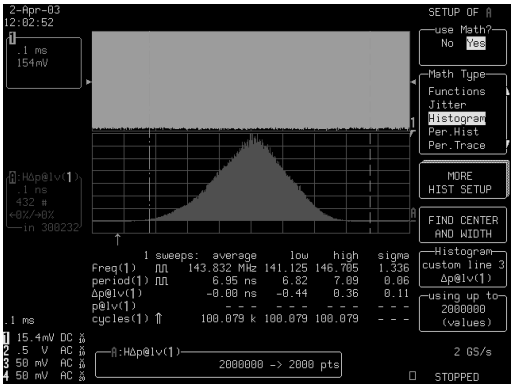

Figure 10: Jitter plot of the DPLL

Table 1: DPLL performance summary

Technology	0.25um CMOS
Power Supply	2.6V
Power Consumption	3.12mW @ 144Mhz
VCO Range	30MHz – 160MHz
Cycle Jitter	130ps
Rms Jitter	60ps

Table 2 compares the normalized results of the proposed DPLL with 4 other PLLs. The first two columns are recent DPLL implementations. The third column is an analog-based PLL that has been implemented in the same technology. The fourth column is an analog-based PLL found in the literature that has excellent performance. For a fair comparison, the power has been normalized to frequency (mW/MHz),, the area has been normalized to the technology (assuming square reduction of area with technology), and jitter has been normalized to the square root of the power consumption. The figure-of-merit (FOM) used to compare all the PLL architecture is given as:

$$FOM = \left[\frac{area(mm^2)}{(tech/0.25)^2}\right]\left[\frac{mW}{MHz}\right]^{1.5}\left[jitter(ps)\cdot\sqrt{mW}\right]^2 \quad (3)$$

Since the most important parameter is jitter, its normalized factor is squared in (3), followed by power with a power of 1.5, and finally area that has only a linear term (since it is least important).

Table 2: Comparison of proposed DPLL with others

	Proposed	[1]	[2]	APLL	[6]
Area	1	2.79	1.113	3.43	0.64
Pwr	1	9.23	12.11	2.30	5.77
T_L	1	0.61	0.085	3.33	2.40
Jitter	1	1.52	3.324	1.22	1.03
FOM	1	182.3	518.8	17.74	9.46

As Table 2 shows, the proposed DPLL outperforms recent DPLL implementations and has comparable performance to analog-based PLL implementations with less power consumption. Less power consumption is in part due to the elimination of overhead analog circuitry.

6. Conclusions

A compact, low-power low-jitter digital PLL has been presented. The traditional analog loop filter was replaced by a digital filter. Fast lock was enabled by a binary locking algorithm. Low-jitter was enabled by the fact that much of the sensitive analog circuitry has been digitised and an adaptive loop bandwidth algorithm has been implemented. Low-power has also been achieved by circuit optimisations in the iDAC and by eliminating much of the power consuming analog circuitry. The proposed DPLL has been demonstrated to outperform recently published DPLLs and have comparable performance to analog-based PLLs.

7. References

[1] I. Hwang, et. al, "A Digitally Controlled Phase-Locked Loop with a Digital Phase-Frequency Detector for Fast Acquisition," *IEEE JSSC*, Oct. 2001, vol. 36, pp. 1574-1581.

[2] C. Chung, C. Lee, "An All-Digital Phase-Locked Loop for High-Speed Clock Generation," *IEEE JSSC*, Feb 2003, vol. 38, pp. 347-351.

[3] A. Bellaouar and M. Elmasry, *Low-Power Digital VLSI Design*, Boston: Kluwer Academic Publishers, 1995.

[4] Tien-Yu, W. et. al., "A Low Glitch 10-bit 75MHz CMOS Video D/A Converter," *IEEE JSSC*, vol. Jan. 1995, pp. 68-72.

[5] Maneatis, J., "Low-jitter and process independent DLL and PLL based on self biased techniques," *ISSCC*, 1996, pp. 130-131.

[6] Ahn, H. and Allstot, D., "A low-jitter 1.9-V CMOS PLL for UltraSPARC microprocessor applications," *IEEE JSSC*, vol. 35, Mar 2000, pp. 450-454.