Jitter Analysis of a Mixed PLL-DLL Architecture

Md. Sayfullah, Barth Roland, Arpad L. Scholtz
Vienna University of Technology
Institute of Communication and Radio Frequency Engineering
Gusshausstrasse 25/389, A-1040 Vienna, Austria
Qimonda AG, Munich, Germany
Email: md.sayfullah@qimonda.com

Abstract — This paper presents the jitter analysis of a mixed mode phase locked loop (PLL) - delay locked loop (DLL) architecture. According to the jitter type, this model can be used as pure PLL or pure DLL or a mixed PLL-DLL. It is observed that mixed mode PLL-DLL architecture can combine the advantage from both PLL and DLL to reduce jitter.

Index Terms — PLL, DLL, mixed PLL-DLL, jitter, jitter transfer function.

I. INTRODUCTION

ADVANCES in integrated circuit (IC) fabrication technology along with innovative circuit design techniques have led to very high-speed digital systems. As clock frequency is increasing, jitter analysis is becoming paramount important for good PLL/DLL design. A phenomenon known as jitter accumulation makes PLLs more susceptible to power-supply and substrate noises [1], [3], [5]. In cases where a significant amount of noise-generating digital circuitry is present on the same chip, DLLs are preferred because any jitter created by the on-chip noise is completely corrected when a clean reference clock edge arrives at the input of DLL [7], [9], [10]. A clean input reference clock might not be available for some high-speed application in order to reduce the cost. And if the reference clock itself might have significant jitter, the utilization of a DLL does not always guarantee superior jitter performance compared to a PLL.

Phase transfer function of a PLL typically exhibits low-pass filter characteristics where as a DLL exhibits all-pass filter characteristics. Therefore, PLL loop can filter high frequency jitter on the reference clock and on the contrary, DLL does not filter out any noise on reference clock. Moreover phase transfer function of a DLL reveals high-frequency jitter peaking, making it unsuitable for application where there is a significant amount of high-frequency jitter associated with the input reference clock.

Considering all these pros and cons of PLL and DLL, this paper presents a mixed PLL-DLL architecture and its’ jitter analysis. Section II provides a general background on PLL and DLL. Section III gives a detail z-domain model of mixed PLL-DLL architecture and its’ simulation result. Section IV concludes with a summary.

II. BACKGROUND

A. PLL

Commonly used PLLs are 2nd order feedback system that generates a clock signal whose output phase is aligned with respect to the phase of an input reference clock. Since phase is the integration of frequency, once the phases are aligned, both phase and frequency are “locked”. Block diagram of a commonly-used charge pump based PLL is presented in figure 1, comprised of a voltage-controlled oscillator(VCO), a phase-frequency detector (PFD), a charge pump (CP) and a loop filter (LF).

![Fig. 1. Charge-pump PLL block diagram](image)

Since PLL is a sampled system, a continuous-time approximation is only valid up to frequencies that are much lower than the reference clock frequency and if the loop bandwidth is much lower than the sampling frequency [11]. Where as z-domain model is more accurate and it also reveals the sampled nature of PLL system. Figure 2 represents z-domain model of the above PLL.
The CP and LF combined transfer function is replaced by its equivalent discrete-time model LF\( (z) \), and \( K_{vco} \) is the VCO gain. For a given input reference clock period \( T_{ref} \) and charge-pump current \( I_{CP} \), combined transfer function LF\( (z) \) becomes

\[
LF(z) = \frac{I_{CP} T_{ref} R(z - \beta)}{2 \pi (z - 1)}
\]

Where, \( \beta = \exp(-\frac{T_{ref}}{RC}) \)

Open-loop transfer function is then

\[
LG(z) = \frac{LF(z)K_{vco}}{(z - 1)}
\]

Input phase transfer function (PTF)

\[
\frac{\phi_{out}}{\phi_{ref}} (z) = \frac{LG(z)}{1 + LG(z)} = \frac{LF(z)K_{vco}}{z - [1 - LF(z)K_{vco}]}
\]

Output phase transfer function

\[
\frac{\phi_{out}}{\phi_{ref}} (z) = \frac{1}{1 + LG(z)} = \frac{z - 1}{z - [1 - LF(z)K_{vco}]}
\]

From Fig. 2, it is clear that input phase transfer function and input noise transfer function (NTF) is the same. Z-domain PLL model simulation result shows the following input and output noise transfer function.

However z-domain model is only valid up to half of the reference clock frequency due to the sampled nature of the system.

### B. DLL

A DLL is also a feedback system and it uses a voltage-controlled delay line (VCDL) instead of VCO to generate an output clock that is a delayed version of the input clock. The delay through the VCDL is a fixed fraction (often 50% or 100%) of the input clock period. While there are several ways to implement a DLL, figure 4 presents a block diagram of a commonly-used charge pump based DLL topology.

For DLL, transfer function LF\( (z) \) becomes

\[
LF(z) = \frac{K_{cp} z}{(z - 1)}
\]

Where \( K_{cp} = \frac{I_{CP} T_{ref}}{2 \pi C} \)

Based on the above expression for LF, input-to-output phase transfer can then be represented by the following expression

\[
\frac{\phi_{out}}{\phi_{in}} (z) = \frac{(1 + K_{VCDL} K_{CP}) z - 1}{z[z - (1 - K_{VCDL} K_{CP})]}
\]

Output noise transfer function

\[
\frac{N_{out}}{N_{in}} (z) = \frac{z - 1}{z - (1 - K_{VCDL} K_{CP})}
\]
Plotting the frequency response of equation (6), figure 6 reveals that the input-to-output phase transfer function has an all pass characteristics, which implies that high frequency jitter on the reference clock passes straight through to the output without being filtering. Actually it is even worse. The high frequency jitter on the reference clock is slightly amplified for frequency higher than

$$\frac{1}{2\pi T_{\text{ref}}} \ln \left( \frac{1}{1 + K_{\text{VCDL}} K_{\text{CP}}} \right),$$

which is the frequency of the zero in equation (6). This jitter peaking is inevitable in this type of DLL design, because for positive values of $K_{\text{VCDL}} K_{\text{CP}}$, the frequency of zero, $(1/(1+K_{\text{VCDL}} K_{\text{CP}}))$, is always larger than the frequency of the pole, $(1-K_{\text{VCDL}} K_{\text{CP}})$, in equation (6).

III. MIXED PLL-DLL ARCHITECTURE

![Fig. 7. Mixed PLL-DLL topology. If $\mu=0$, it is a PLL mode, $\mu=1$ means DLL mode and $0<\mu<1$ is the mixed mode.](image)

This circuit uses a phase mixing interpolator to configure the loop as a PLL, DLL or a mixture of the two. If the on-chip supply is noisy, the high-bandwidth DLL-mode is preferred to avoid supply noise accumulation. On the other hand, if the reference clock is noisy, the low-bandwidth PLL-mode is preferred to take the advantage of low-pass filtering characteristics of the loop. When both noise sources, whose relative amount not be known a priori, are present, a mixed-mode may yield an optimum bandwidth setting to minimize output jitter [6].

Phase mixing interpolator as the first delay element mixes signal energy between output clock and a buffered reference clock based on a 3-bit digital code $\mu[2:0])$. A mixing weight of 0 corresponds to PLL mode without any reference clock injection (VCO-mode), a mixing weight of 1 corresponds to DLL-mode with full reference clock injection strength (VCDL-mode), and four intermediate mixing weight (0.2, 0.4, 0.6, 0.8) correspond to mixed-mode operations with relative weight strengths between $\text{Clk}_{\text{in}}$ and $\text{Clk}_{\text{out}}$ (mixed VCO/VCDL-mode).

Z-domain representation of a mixed type PLL-DLL structure is shown in figure 8.

![Fig. 8. Z-domain representation of a mixed mode PLL-DLL.](image)

In figure 8, $I$ is representing the injection strength of reference clock whose phase is represented by $\phi_{\text{in}}$, and $(1-I)$ is the injection strength of $\text{Clk}_{\text{out}}$.

Input phase transfer function

$$\frac{\phi_{\text{out}}}{\phi_{\text{in}}} = \frac{I + LF(z)K_{\text{VCDL}}}{z + LF(z)K_{\text{VCDL}} - (1-I)}$$

(8)

Where

$$LF(z) = \frac{I_{CP}T_{\text{ref}}R(z - \beta)}{2\pi(z - 1)}, \quad \beta = \exp\left(-\frac{T_{\text{ref}}}{RC}\right)$$

Output noise transfer function

$$N_{\text{out}} = \frac{z}{z + LF(z)K_{\text{VCDL}} - (1-I)}$$

(9)

The simulation results of the PTF and NTF across different mixing weights are shown in the following figures.
Input phase transfer function plot demonstrate that loop bandwidth of this mixed PLL-DLL can be tuned over a wide range of frequencies, from the lower PLL bandwidth all the way to half the reference clock frequency by changing the relative mixing weight between Clk_in and Clk_out. The jitter peaking in the PLL mode is less than 0.5dB with a large damping ratio. Some jitter peaking is also observed at high frequencies in the DLL mode. This jitter peaking can be reduced by linearly scaling the loop filter resistor with Clk_out injection strength, (1-\(I_I\)), such that the zero is removed in DLL mode.

Before we compare the supply noise response of a PLL and a DLL, we must understand how the supply or substrate noise gets into a PLL or DLL. If the supply or substrate reference level changes, a change on the delay of the delay elements in the VCO in a PLL, or the VCDL in a DLL, occurs. A performance measurement called supply sensitivity is usually expressed in a normalized percentage of delay change per percentage of supply change (% delay / % volt). The delay modulation of the delay elements in the VCO/VCDL requires the use of delay element design with good supply noise rejection. A PLL usually has higher supply noise sensitivity than a DLL using the same delay element because a change in supply voltage results in a change in VCO’s output frequency, which integrates the phase error within the feedback loop until the loop’s correcting action takes effect.

Since PLL typically has a low bandwidth in order to guarantee stability, the phase error accumulation can result in a large amount output jitter under a noisy supply. On the other hand, a change of a VCDL’s supply voltage results in only a delay change occurring once through the delay line. There is no phase error accumulation within the loop due to a fresh reference clock edge that feeds into the VCDL every reference cycle. Therefore a PLL has more supply-induced jitter than a DLL\[10\].

**IV. CONCLUSION**

Two of the most significant noise source in today’s digital circuitry are –
- Input reference clock noise
- Supply- or substrate-induced noise.

PLL or DLL alone can not attenuate these two noises at the same time. From simulation result, it is shown that a mixed PLL-DLL architecture can attenuate both noises with appropriate mixing weight.

**ACKNOWLEDGEMENT**

The author wishes to acknowledge the support of all members in Qimonda concept engineering team.

**REFERENCES**

