Title
Clock and Data Recovery Loops: A Frequency Domain Approach

Author
Fayazi, Mohammadhasan

Publication Date
2016

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

Clock and Data Recovery Loops: A Frequency Domain Approach

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Electrical Engineering

by

Mohammadhasan Fayazi

2016

Abstract of the Thesis

Clock and Data Recovery Loops: A Frequency Domain Approach

by

Mohammadhasan Fayazi

Master of Science in Electrical Engineering

University of California, Los Angeles, 2016

Professor Asad A. Abidi, Chair

While being frequency compact and easy to implement, Non-Return to Zero (NRZ) encoded data does not contain any energy at its clock frequency which makes the clock extraction impossible using any kind of Linear Time Invariant (LTI) operations. Therefore, Clock Data Recovery circuits (CDRs) have an inherent non linear recovery process. In this work we present a frequency domain analysis of the mechanisms leading to the energy generation at clock frequency for NRZ clock data recovery systems. We also propose a frequency domain analysis which is applicable to both Bang-Bang and linear loops. We show the theory results match the measurements very well.

The thesis of Mohammadhasan Fayazi is approved.

Chih-Kong Ken Yang

Sudhakar Pamarti

Asad A. Abidi, Committee Chair

University of California, Los Angeles

2016

To my parents . . .

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to express my highest gratitude to Professor Abidi. Regardless of how busy he was, he always allocated the time to patiently discuss my research and guide me with his enlightening comments and viewpoints. His emphasis on deep understanding of concepts and utilizing first principles to develop the most complicated theories is a great lesson I have learnt from him.

I would also like to thank my talented fellow lab mate, Hao Xu. The loop analysis in chapter four is based on one of his unpublished works. He kindly helped and discussed various aspects of this work with me. I am very blessed to be in this research group so I can enjoy learning from a great advisor and working with intelligent lab mates.

I owe all my achievements to my parents, who have selflessly supported and loved me.

Vita

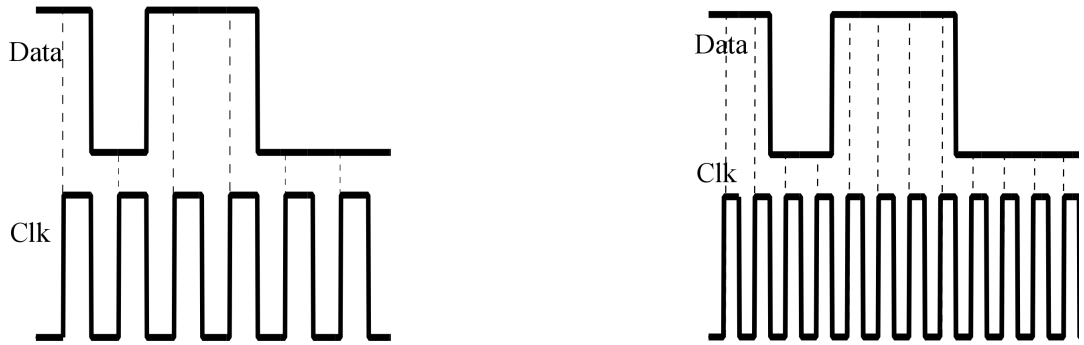
2010–2014 B. Sc. In Electrical Engineering, Sharif University of Technology, Tehran, Iran

CHAPTER 1

Introduction

1.1 Motivations

The advantages of Digital Communication over its Analog counterpart have been stated in [1] and [2]. This type of communication is employed in both wireline and wireless systems. While the latter needs a carrier frequency, the former does not and is referred to as "Serial Communication" since digital bits are transmitted one by one. In order to decode the transmitted message a synchronous clock with the incoming data is required [3]. Figure 1.1 shows this necessity. As we can see, although data waveform is identical in both cases, received messages in Figure 1.1 (a) and (b) are 101100 and 110011110000, respectively.



(a) Received message is 101100 (b) Received message is 110011110000

Figure 1.1: Data sampled by different clocks

There are various standards for data encoding in serial communication among which

Pulse Amplitude Modulation (PAM) and especially 2-PAM, also known as Non Return to Zero (NRZ), and 4-PAM have become very popular, because of their bandwidth efficiency which proves to be desirable when dealing with bandlimited channels [4–6]. Interestingly these encoding schemes do not have any energy at the clock frequency (see section 3.1). Therefore, clock cannot be extracted with any Linear Time Invariant (LTI) operation [7].

In this work we present a mathematical model for the processes used to recover clock from random NRZ data. These methods have been designed based on common sense but according to our knowledge a rigorous analysis of them is lacked.
In addition these loops are mostly designed by reliance on time domain simulations, which is extremely time consuming. There have been efforts for a frequency domain analysis for Phase Locked Loops or CDRs [8–13]. These analyses mostly do not include the effect of random sampling of phase errors in a CDR or the noise introduced by it [8–12] and hence are not accurate. [13] discusses the random data transition and the noise associated with it, but this discussion is limited to non-linear phase detectors. In addition, it relies on graphical solution for the phase detector gain calculation and does not offer a closed form solution. It has also neglected the deterministic jitter effect on the phase detector gain.
In this work we present a frequency domain analysis that is capable of producing very accurate results with minimal computational cost, and also provide closed form expressions for gain calculation in non-linear loops. This analysis allows loop design without recourse to time domain simulations.

1.2 Organization

This work is organized as follow: In chapter 2 we briefly overview the fundamental blocks and concepts in a wireline link.
In chapter 3 the frequency content of different modulation schemes is investigated and the generation of a tone at desired clock frequency by derivation and rectification is discussed.

Chapter 4 introduces a linearization method for Bang Bang Phase Detectors. It also includes an overview on s-domain analysis of clock and data recovery and phased locked loops. We will explain how the concept of noise bandwidth is applicable to such systems and present formulas for an easy calculation of mean squared phase noise at the output.

Analysis introduced in chapter 4 is verified against silicon measurement in chapter 5. We analytically calculate power spectral density of the output phase noise and its mean squared value and show it closely matches the silicon measurement.

CHAPTER 2

An Overview on Wireline Communication

2.1 A Wireline Link

Figure 2.1 shows a general illustration of the key parts in a wireline link. we will briefly introduce each of these parts in this section.

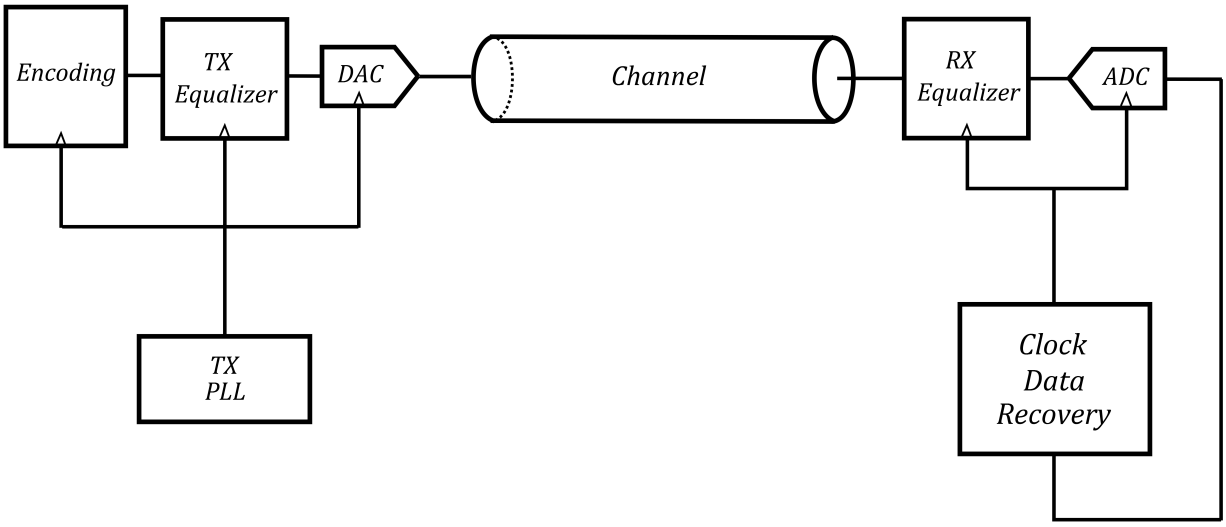
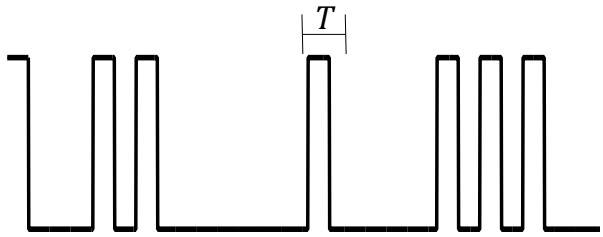


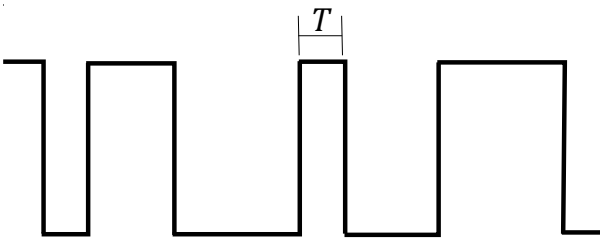
Figure 2.1: A Wireline Link

Digital data needs to be modulated before being sent and this operation is done by the *Encoding* block. There are numerous encoding methods [5, 14, 15], the most frequently used of which are 4-PAM, 2-PAM, and Return to Zero (RZ) (Figure 2.2). A proper choice of encoding scheme heavily depends on the application, as each of them have their own

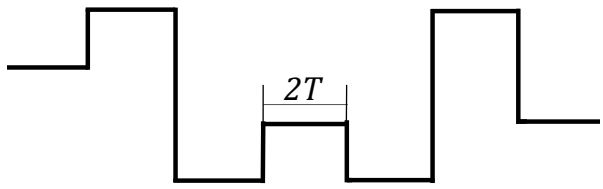
advantages and disadvantages [16].



(a) RZ waveform



(b) NRZ waveform



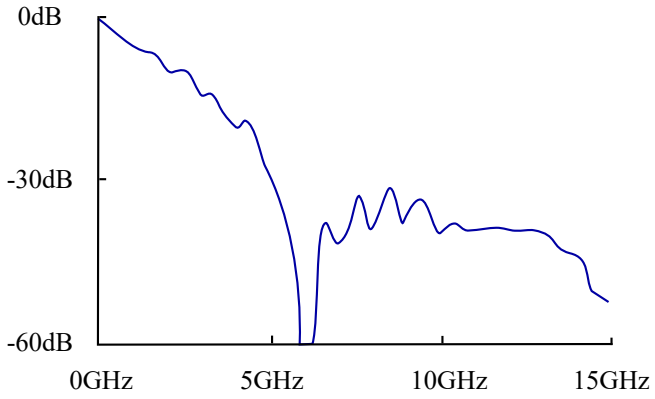
(c) 4-PAM waveform

Figure 2.2: Various encoding schemes

As shown in Figure 2.2, in order to obtain the same symbol rate, the symbol period in a 4-PAM system can to be twice that for 2-PAM and RZ systems. Thus, the required bandwidth is two times less for 4-PAM. However, having the same voltage headroom, 4-PAM is more susceptible to voltage noise as there are more possible voltage levels.

Channel is the media through which signal propagates. It can be either optical fiber or electrical wire. A channel can be crudely modeled by a low pass filter. Figure 2.3 shows an

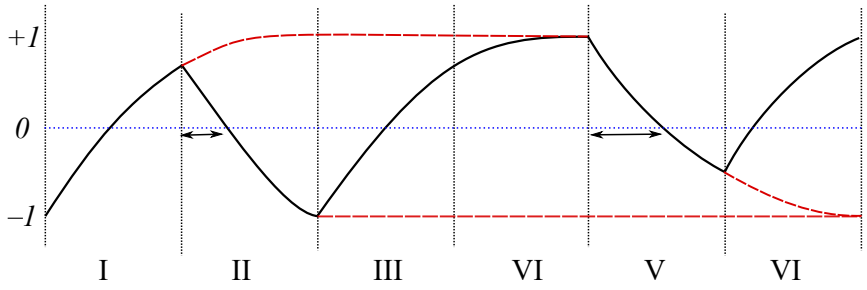
example channel (a backplane with 5mm via stub) frequency response, which is borrowed from [17].



5mm via stub back plane channel frequency response from [17]

Figure 2.3: An example channel frequency response

As Figure 2.3 shows, channels have finite bandwidths, which, in time domain, translates into finite speeds . As a result, in high speed applications pulse tails have not died out by the time the next pulse arrives. This situation is shown in Figure 2.4.



Dashed lines show the trace if it was not disturbed by other pulses.

Figure 2.4: ISI generation mechanism

This phenamenon, which is called *Inter Symbol Interference (ISI)*, causes the channel response at each time to be determined by the transmitted bit at that time as well as effects

from the other (mostly previous but in some cases even future) transmitted bits. In Figure 2.4, the second and fifth intervals were expected to be the same (because both are a "1" to "0" transition), but it could be readily seen that they are not and the zero crossing has changed a result of different effects from the previous bits.

Severe ISI can result in link failure. In order to mitigate this problem, *equalizers* are employed in transmitter and (or) receiver to compensate for the channel frequency behavior by implementing the inverse of its transfer function. They are mostly implemented adaptively, as the channel response is not precisely known beforehand. There are various options for these blocks, two of which can be found in [4, 18].

DAC is required because digitally encoded data needs to be converted into analog voltage so as to be transmittable over the channel. In some applications the clock might also be sent along with the data over a separate path to ease the timing recovery process.

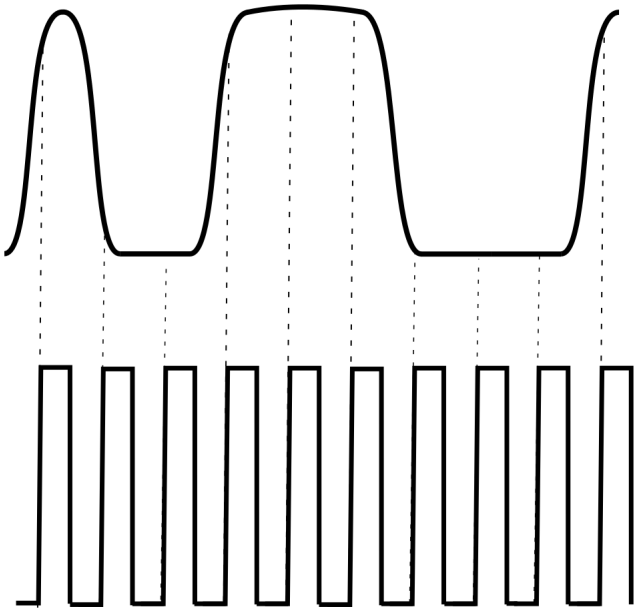


Figure 2.5: ISI experienced data and non-optimum clock

The *Clock and Data Recovery (CDR)* block is responsible for generation of the clock which samples the received waveform (to associate a logic level to it) and drives the digital blocks in the receiver. Therefore, it needs to be synchronized with the incoming data. The received data has experienced ISI and, therefore, analog voltages associated with different logical levels have become close to each other and transitions smooth out (Figure 2.5). So as to determine the transmitted bit correctly we need to sample the analog received voltage with a proper clock. The optimum sampling point is exactly in the middle of each transmitted bit, where the voltage waveform is locally flat, so if the sampling position changes slightly the sampled voltage does not change severely. This optimum sampling implies, other than the right frequency, the recovered clock needs to have correct phase. It is notable that even if the clock is sent the phase needs to be adjusted. The reason is that albeit the received clock has the correct frequency, it might have a shifted phase with respect to the data as they have traversed different paths [19].

2.2 Wireline Communcation Terminology

In this section we introduce definitions and concepts used in wireline communication. These definitions are mostly taken from [20].

Unit Interval (UI): It is defined as the time associated to a single transmitted bit. All time references are measured and reported with respect to UI. In figure 2.2 (a) and (b) T is one UI and in Figure 2.2 (c) one UI is $2T$.

Jitter: Jitter is the time domain effect of phase noise. As it is discussed in [21] jitter could be viewed as phase noise sampled at zero crossings. It is quantified with different definitions, such as "Timing Jitter", "Period Jitter", and "Accumulated Jitter". Shimanouchi in [22] has clearly identified these terms. Later in this work we refer to the timing jitter simply as "jitter".

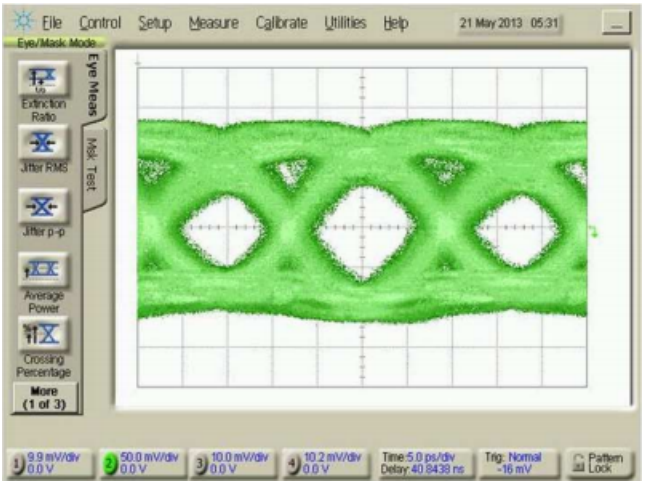
Jitter sources can be divided into several categories. Random noise sources in the system such as thermal or flicker noise cause *Random Jitter*. On the other hand *Deterministic Jitter*

and *Periodic Jitter* are generated by non random sources. Periodic jitter is a result of any kind of coupling to periodic sources in the system such as clock feedthrough. Unlike periodic jitter, deterministic jitter is not necessarily periodic. A bandlimited channel creating ISI causes deterministic jitter, because depending on the previously transmitted random data, zero crossings might change [23]. This phenomena is observable in Figure 2.4. The zero crossings have changed in different intervals because of ISI.

Bit Error Rate: Bit Error Rate (BER) is a gauge of the receiver's decision accuracy. As we sample a random data due to noise and Jitter we might err in associating the correct bit to the sampled analog voltage. Acceptable BER is very small and varies from application to application. For instance, it is 10^{-12} for SONET standard.

A useful tool for characterizing a channel is the *Eye Diagram*. If we overlay several UIs we will get a plot similar to Figure 2.6. If data is chosen truly randomly and the number of overlaid UIs is large, its covering approaches all possible scenarios in the channel. Eye Closure and Eye Opening are closely related to ISI severity and dictate the receiver sensitivity. *Eye Opening* can be tens of millivolts for high speed I/O links. Figure 2.6 shows an example of eye diagram measurement for a 60GB/s NRZ data from [24]. As we can see the the eye opening is 100mV.

It is possible to plot BER versus timing error and voltage error simultaneously. We discussed how deviation of sampling time from the center of eye diagram can lead to error. This plot, an example of which is shown in Figure 2.7, depicts how BER changes (for fixed timing offsets) with an offset in the threshold voltage. It can be seen that even if samples are taken at the eye center, offset in the threshold voltage will lead to error. This graph gives the time and voltage margins for a desired BER.



The eye diagram measurement for 60GB/s NRZ data [24]

Figure 2.6: An example of eye diagram

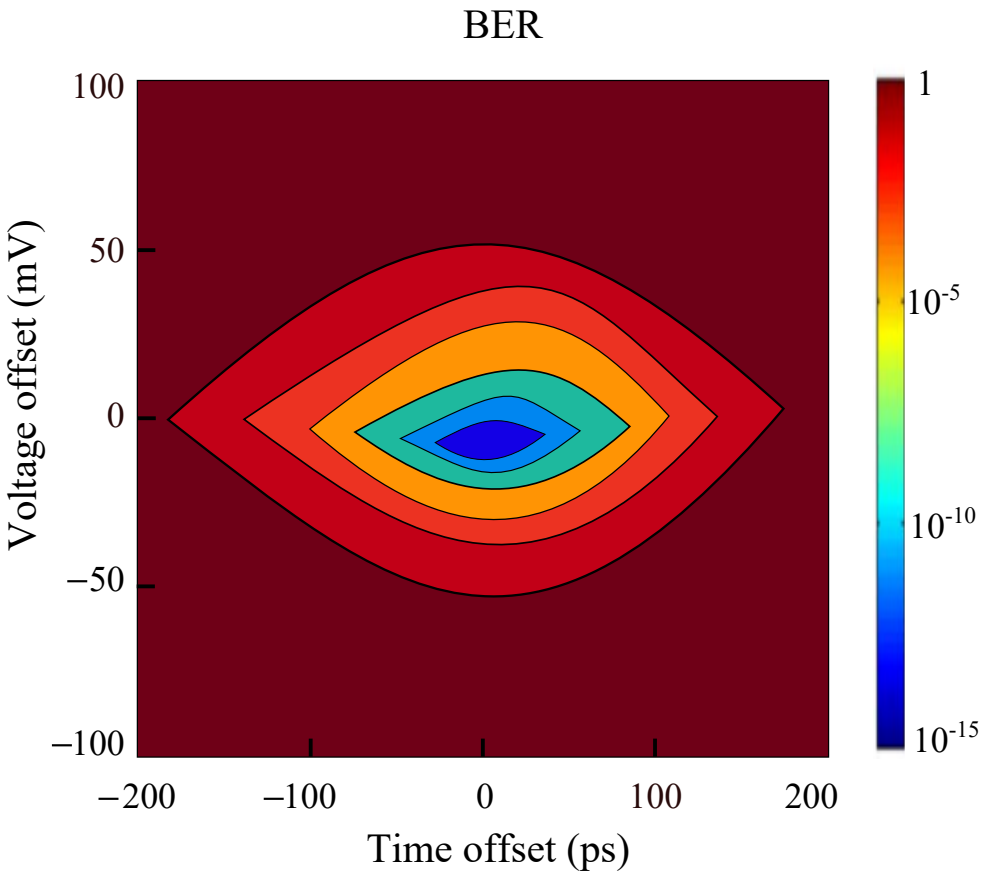


Figure 2.7: An example of BER vs time and voltage offsets

CHAPTER 3

Clock Data Recovery from NRZ Data

3.1 Power Spectral Density of a Random Bitstream

We pointed out, without proof, that NRZ data does not contain any energy either at clock frequency or any of its harmonics. In this section we derive the Power Spectral Density (PSD) of different data encoding schemes.

A bitstream could be represented mathematically as follows:

$$x(t) = \sum_{k=-\infty}^{\infty} a_k p(t - kT) \tag{3.1}$$

where $p(t)$ is the modulating pulse and $T = \dfrac{1}{f_{clk}}$ is 1 UI, as was defined in Chapter 2. The sequence a_k is data and is a Wide Sense Stationary (WSS) random process. For NRZ data $p(t)$ ideally spans 1 UI, while for RZ modulation it is shorter than 1 UI, which means the transmitted waveform goes back to zero after each bit. According to [2] the PSD of $x(t)$ is

$$S_x(f) = \frac{1}{T}|P(f)|^2 \sum_{n=-\infty}^{\infty} R_a(n) e^{-j2\pi nfT} \tag{3.2}$$

In this expression $P(f)$ is the Fourier Transform of the modulating pulse, $p(t)$, and $R_a(n)$ is defined as the auto-correlation between a_k and a_{k+n}.

$$R_a(n) = E(a_k a_{k+n}) \tag{3.3}$$

We note that the existence of $R_a(n)$ is guaranteed by the WSS assumption on a_k. For the case where a_k's are Independent Identically Distributed (i.i.d.) with average of m_a and standard deviation of σ_a, another more insightful formulation of (3.2) can be derived which

is also presented in the same reference.

$$\begin{aligned} S_x(f) &= \frac{1}{T}|P(f)|^2(\sigma_a^2 + \sum_{n=-\infty}^{\infty} m_a^2 e^{-j2\pi nfT}) \\ &= \frac{\sigma_a^2}{T}|P(f)|^2 + \frac{m_a^2}{T}|P(f)|^2(\sum_{n=-\infty}^{\infty} e^{-j2\pi nfT}) \end{aligned} \tag{3.4}$$

If we use the fact that $\sum_{n=-\infty}^{\infty} m_a^2 e^{-j2\pi nfT} = \sum_{n=-\infty}^{\infty} \delta(f - \frac{n}{T})$ we can further simplify (3.4)

$$S_x(f) = \sigma_a^2 f_{clk}|P(f)|^2 + (m_a^2 f_{clk}) \sum_{n=-\infty}^{\infty} |P(nf_{clk})|^2 \delta(f - nf_{clk}) \tag{3.5}$$

It is noteworthy that in (3.5) there is the potential for impulses to appear at clock frequency and its harmonics. 3.5 also tells us that these impulses will be evident if $|P(nf_{clk})|$ is non-zero for $n = 1, 2, ...$ and data bits have a non-zero average (i.e. $m_a \neq 0$). For almost all practical cases $p(t)$ is a square pulse with duration T_p, which is set by the modulation as shown in Figure 3.1. The Fourier Transform of $p(t)$, $P(f)$, in this case would be a Sinc function with its nulls at multiple integers of $\frac{1}{T_p}$.

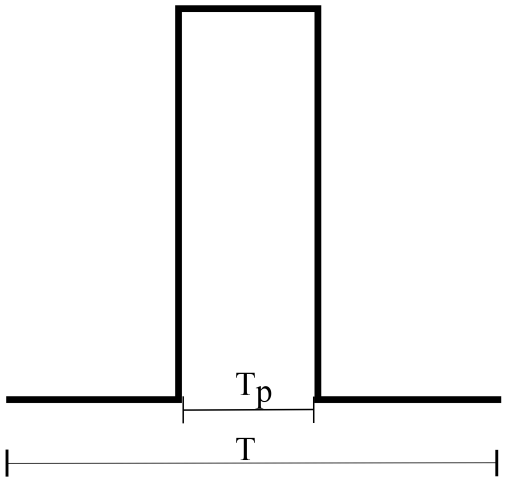
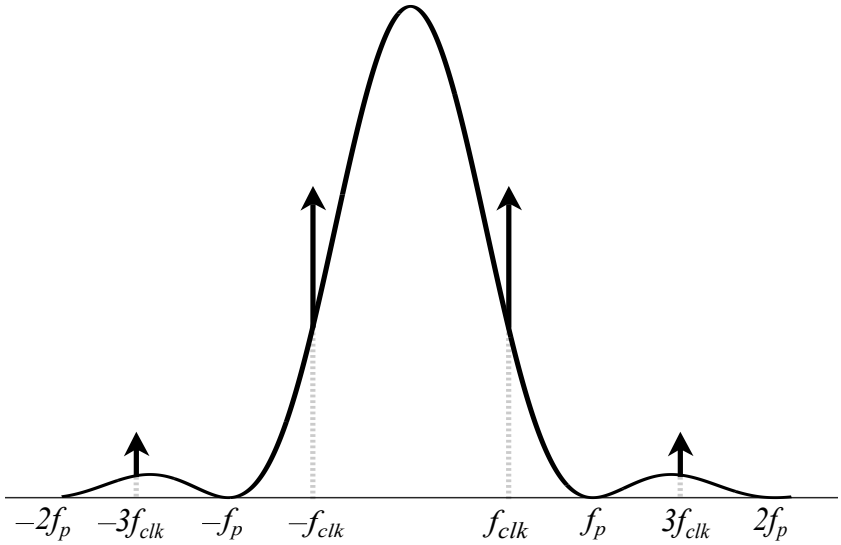


Figure 3.1: Modulating Pulse

In RZ data, the pulse duration is less than a bit period and, hence, the first null will be at a higher frequency than the clock frequency. This means $P(\frac{1}{T}) = P(f_{clk}) \neq 0$ and the random data PSD contains impulses at clock harmonics. Figure 3.2 shows the spectrum of a RZ waveform (For RZ modulation, conventionally, $a_k = +1$ or $0 \Rightarrow m_a \neq 0$).
On the contrary, because $T_p = T$ for NRZ data, $P(f)$ has nulls exactly at clock harmonics which suppresses the impulses. NRZ data PSD is shown in Figure 3.3.

The only difference between 2-PAM and 4-PAM is that a_k takes only two different values in a 2-PAM system, while it can take four values in a 4-PAM system. However, 4-PAM still has a zero average but its variance might be different from that of 2-PAM. This tells us that



Spectrum of a RZ waveform with $T = \frac{T_p}{2}$. Clock even harmonics are suppressed.

Figure 3.2: RZ data PSD

their PSD will be similar, only with different σ_a, and without any energy at clock frequency.

3.2 Clock Data Recovery Architectures

In general, a selective bandpass filter can extract a specific frequency. This is similar to our clock recovery problem. If the incoming data has non-zero energy at clock frequency, we need a high Q filter or a Phase Locked Loop (PLL) to extract that component [25]. However, a PLL does not lock if random data is given to it as the input, and an LTI filter tuned to the clock frequency does not pick up any energy at f_{clk} if an NRZ data is fed to it. This suggests some non-linear action to generate energy at clock frequency. We have demonstrated before that a random bitstream will contain a tone at clock frequency if the modulating pulse has non-zero frequency content at f_{clk} and the bits have non-zero mean. None of these conditions

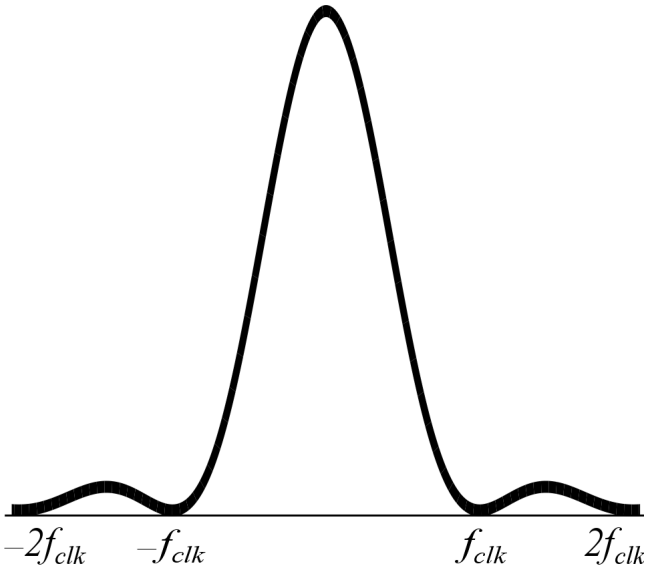


Figure 3.3: NRZ data PSD

are met for NRZ data. Taking derivative from $x(t)$ will give us

$$x'(t) = \sum_{k=-\infty}^{k=\infty} (a_k - a_{k-1})\delta(t - kT) \tag{3.6}$$

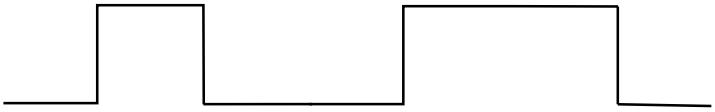
As we can see the modulating pulse has become an impulse, which has a white spectrum and, therefore, meets the first condition. Still, obtained by a linear operation x' does not have any spectral line at f_{clk} due to its zero mean ($a_k - a_{k-1}$ is zero or takes $+2$ and -2 with equal probability[1]). However, if we pass x_{der} through a rectifier it will create a non-zero average for the bits and, thus, a spectral component at clock frequency. This operation could be done in analog or digital domain [26] and the result fed to a PLL or a high Q filter. Figure 3.5 shows these two methods. Besides rectified derivative [27], squaring has also been used as a non-linearity for clock generation [7].

Usually a PLL is preferred over a conventional filter since it does not rely an advance knowledge of the clock frequency, and does not need high quality passive[2].

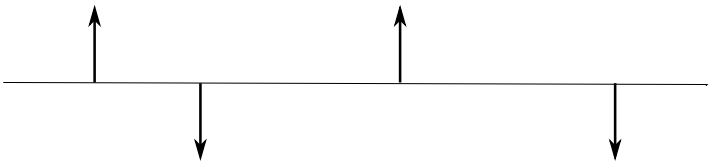
Another approach for clock recovery systems is to drive the phase of the recovered clock

[1] a_k takes $+1$ and -1 with equal probability for NRZ data

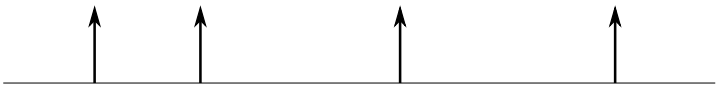
[2] A sharp filter bandpass filter would need high Q passive elements



(a) The incoming data. Zero average and $P(f_{clk}) = 0$



(b) The incoming data derivative. Zero average but $P(f_{clk}) \neq 0$



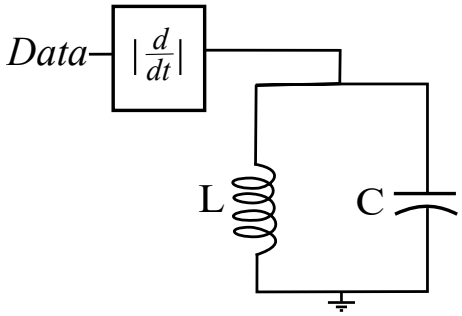
(c) The incoming data rectified derivative. non-zero average and $P(f_{clk}) \neq 0$

Figure 3.4: Generation of a tone at clock frequency by derivation and rectification

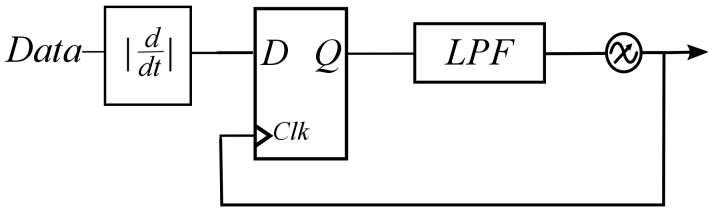
to the optimum sampling point. In such systems the derivatives of the voltage at sampling points operate as the error signal for a feedback loop, and the loop behavior diminishes this error. In that situation the sampling edges strobe the waveform exactly at the optimum point which is its maxima or minima [28].

3.3 Phase Detection for Random Data

The reason a PLL fails to lock to NRZ data is that conventional Phase Detectors (PDs), an example of which is shown in 3.6 (a), do not operate correctly for random data. Figure 3.7 illustrates an example where two late random data waveforms. If the shown clock and



(a) By passive filtering. L and C must be chosen so $\omega_{clk} = \frac{1}{\sqrt{LC}}$



(b)

(b) Using a PLL as filter

Figure 3.5: Clock recovery from data rectified derivative by filtering

data waveforms are connected to a conventional PD the output will be different, while it was expected to be the same. The problem arises because the phase detector tries to infer timing information from input levels, which could have been done if data was deterministic (where levels and timing frames are tied together). Still, as far as random data in concerned, such connection does not exist.

If we just swap the role of clock and data (Figure 3.6(b)) the problem will be solved. As we can see, in both example waveforms a zero level on clock is sampled, which corresponds to a late decision. The reason is that this circuit concludes phase information from transition edges of data, which are starting points of timing frames, and the clock, a deterministic signal, levels.

Another way to look at these two circuits is by focusing on a D Flip-Flop (DFF) function.

We know DFFs sample their input at clock rising or (and) falling edges. Therefore, the output of the circuit (a) in Figure 3.6 is $Data \times r(\frac{d}{dt}Clk)$, where:

$$r(x) = \begin{cases} x & x > 0 \\ 0 & \text{o.w} \end{cases} \tag{3.7}$$

whereas in circuit (b) the output will be $Clk \times r(\frac{d}{dt}Data)$. We can clearly see the first circuit only performs a linear operation on the incoming data, but the second half wave rectifies $\frac{d}{dt}Data$, which will create a tone at the clock frequency.

(a) A conventional PD

(b) A random data PD

Figure 3.6: A conventional PD and a PD for random data

When the incoming data is sampled by the clock, any misalignment between the sampling edge and the center of eye diagram will degrade BER. This misalignment occurs because of the phase error between the data and clock. Thus, minimizing the phase error is of a great consequence. It is known that the static phase error can be eliminated using a Type-II loop. However, if this clock is used outside the loop to sample data, then due to parasitic delays we cannot guarantee zero static phase error. Therefore, it is desirable to include the data

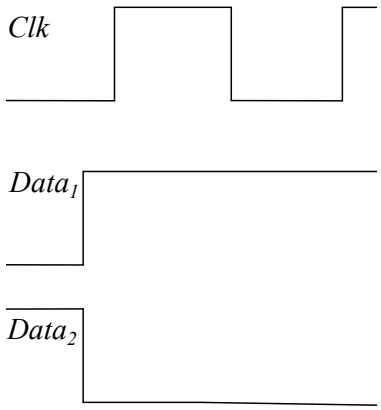


Figure 3.7: An example of two late random data waveforms

sampling inside the loop at the phase detector, so the phase error is forced to be zero by feedback characteristics.

Two widely used phase detectors for CDRs are the one by Hogge [29] and the one by Alexander [30]. As will be discussed, the phase comparison is performed in both of these phase detectors at transitions in incoming data. Both combine rectification, differentiation, and phase detection, which makes them suitable for random data applications. They also sample the input as a part of their operation, which yields the optimum data estimate, without additional parasitic delay.

3.3.1 Hogge Phase Detector

Since this type of phase detector provides an output linearly related to the phase difference, it is very common in CDR loops where output jitter is the main concern. Figure 3.8 shows the implementation of this phase detector as described by Hogge in [29]. When data changes first XOR will produce "1" till clock rising edge comes and the first D Flip Flop makes its output equal to the new data. This pulse width provides a measure of the phase difference between the input data and clock. When the output of first DFF changes second XOR produces "1" till falling clock edge arrives. It is notable that duration of the pulse at the output of second XOR is exactly equal to the time clock is high, which is half clock period for most cases. The difference of the output of two XOR gates $(X - Y)$ is fed as the phase error signal to a loop. We can easily understand the duty cycle of this waveform varies with the phase difference, and think of it as a Pulse Width Modulated (PWM) signal. Given that the loop has enough gain at DC, error will diminish to zero which means the difference between clock rising edge and data transition becomes equal to half of the clock period. This indicates the rising edge happens exactly at the middle of eye diagram and minimizes BER.

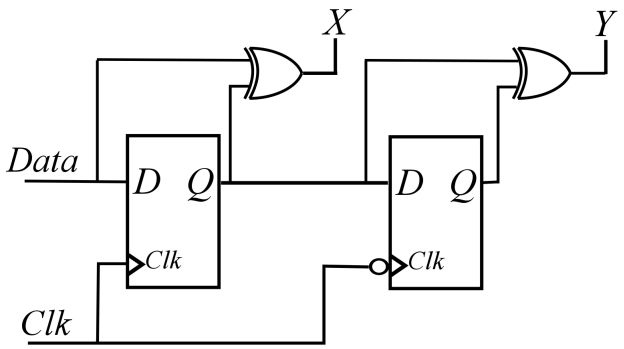


Figure 3.8: Hogge phase detector

3.3.2 Alexander Phase Detector

This phase detector provides a quantized output, based solely on whether the clock is late or early at each data transition. It samples data with twice the clock frequency (using both rising and falling edges) and compares the samples at the falling edge samples with samples at the previous and next rising edge (Figure 3.9). We name falling edge samples E_i and rising edge samples D_i, because in lock, as shown in Figure 3.10, falling edges coincide with data edges and rising edges with the center of eye diagram (Figure 3.10). If E_i is only equal to D_i, it implies the clock is early (D_1, E_1, and D_2 in Figure 3.9). If E_i is the same as D_{i+1} but different from D_i, the clock is late (D_2, E_2, and D_3 in Figure 3.9). The case where $D_i = E_i = D_{i+1}$ means there was no transition in data, so no output is produced (D_3, E_3, and D_4 in Figure 3.9).

The circuit of Figure 3.11 implements the operations discussed above. As was mentioned, we need to sample data using both positive and negative edges, and at each decision instant, in addition to the sample at that time, we need two previous samples. Decisions are made at rising edges which means we need the previous rising edge sample. Q_3 is the previous rising edge sample and Q_4 retimes the falling edge sample so the outputs X and Y change only at positive clock edges. $(X - Y)$ provides the error information.

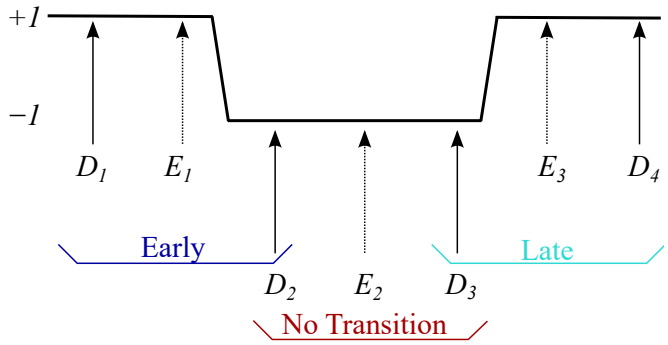


Figure 3.9: Early-Late decision based on sampling

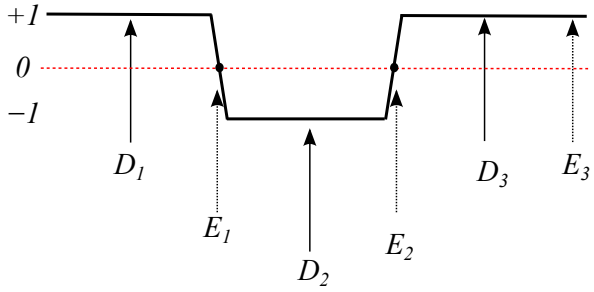


Figure 3.10: Early-Late decision in lock

3.4 NRZ Data Phase Detector (PD) Output Spectrum

In this section we want to investigate the output PSD of a phase detector connected to NRZ data. For the case of deterministic data we know the phase difference, $\Delta\phi$, between input and output always appears at the PD's output. Whereas, in CDRs the phase difference will appear at the PD output if a transition in data happens, otherwise PD's output is 0. We can model this as a random sampling of the phase difference or, in other words, multiplication of phase difference by a random impulse train $w[i]$ (Figure 3.12).

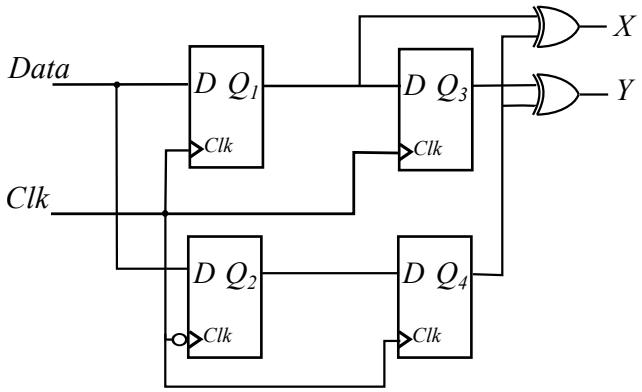


Figure 3.11: Alexander Phase Detector

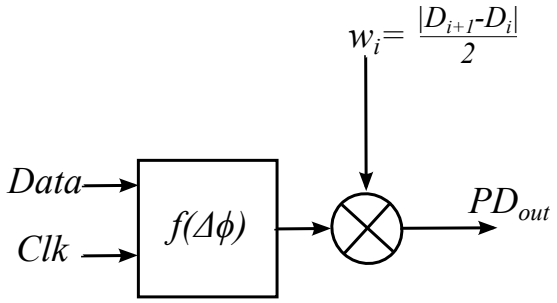


Figure 3.12: Random Sampling of Phase Error

If we assume D_i, the data, takes values of $+1$ and -1, $w[i] = \left|\frac{D_{i+1} - Di}{2}\right|$. With this definition $w[i] = 1$ when $D_i \neq D_{i+1}$ and 0 otherwise. The function $f(\Delta\phi)$ in Figure 3.12 depends on the phase detector architecture. This function for a Bang-Bang PD is $sgn(\Delta\phi)$ and for a linear PD $f(\Delta\phi) = \frac{1}{2\pi}\Delta\phi$.

D_{i-1}	D_i	Transition
-1	-1	$\times$
-1	$+1$	✓
$+1$	-1	✓
$+1$	$+1$	$\times$

Table 3.1: Possible pairs of bits

Consider $PD_{out}[i] = w[i]f(\Delta\phi[i])$, as in Figure 3.12. (3.8)

If D_i is i.i.d. $+1$ and -1 with equal probability, the probability of $w[i] = 1$ which means the probability of a data change from D_i to D_{i+1} by enumeration of all possible pairs of bits that correspond to data transition in Table 3.1 is:

$$p(w[i] = 1) = \underbrace{\frac{1}{2}}_{D_i = +1} \times \underbrace{\frac{1}{2}}_{D_{i+1} = -1} + \underbrace{\frac{1}{2}}_{D_i = -1} \times \underbrace{\frac{1}{2}}_{D_{i+1} = +1} = \frac{1}{2} \tag{3.9}$$

The auto-correlation function of PD_{out} is needed for calculating its PSD. We are assuming PD_{out} is WSS.

$$\begin{aligned} R_{PD_{out}}[n] &= E(PD_{out}[k] \times PD_{out}[k+n]) \\ &= E\Big((w[k]f(\Delta\phi[k])) \times (w[k+n]f(\Delta\phi[k+n])) \Big) \\ &= E(w[k]w[k+n]f(\Delta\phi[k])f(\Delta\phi[k+n])) \end{aligned} \tag{3.10}$$

Because $w[i]$ is a random process, we need to look at power spectral densities. Although $w[i]$ and $\Delta\phi[k]$ might be correlated due to ISI, we prove in Appendix A that this correlation does not change the results and we can treat them as if they were uncorrelated. Therefore:

$$\begin{aligned} R_{PD_{out}}[n] &= E(w[k]w[k+n]) \times E(f(\Delta\phi[k])f(\Delta\phi[k+n])) \\ &= R_{f(\Delta\phi)}[n] \times R_w[n] \end{aligned} \tag{3.11}$$

Because multiplication in time domain translates into convolution in frequency domain

$$S_{PD_{out}}(e^{j\omega}) = S_{f(\Delta\phi)}(e^{j\omega}) * S_w(e^{j\omega}) \tag{3.12}$$

In order to find $S_w(e^{j\omega})$ we need the auto-correlation of $w[i]$.

$$R_w[n] = E(w[k]w[k-n]) \tag{3.13}$$

If $n = 0$, $R_w[n = 0] = E(w[k]w[k])$ which is the same as $E(w[k])$, because $w[i]$ takes only $+1$ and 0. If $|n| > 1$, $w[k]$ and $w[k-n]$ are i.i.d., which means:

$$R_w[n] = E(w[k]w[k-n]) = E(w[k])E(w[k-n]) = E(w[k])^2 = \frac{1}{4} \tag{3.14}$$

For $n = \pm 1$, $w[k]$ and $w[k-n]$ are correlated, because $w[k]$ depends on two consecutive bits. In that case:

$$R_w[n] = E(w[k]w[k-n]) = E(1 \times w[k-n])\Big|_{w[k]=1} + E(0 \times w[k-n])\Big|_{w[k]=0} = \frac{1}{4} \quad (3.15)$$

Therefore,

$$R_w[n] = \frac{1}{4} + \frac{1}{4}\delta[n] \quad (3.16)$$

This auto-correlation function is shown in Figure 3.13.

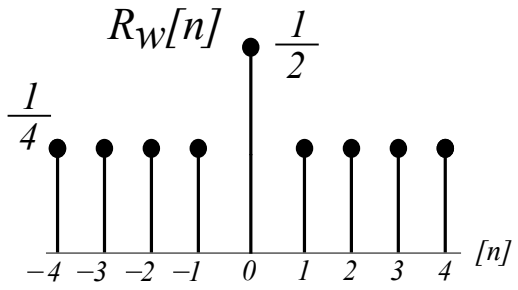


Figure 3.13: auto-correlation of $w[i]$

The PSD is the Fourier Transform of auto-correlation function.

$$R_w[n] \leftrightarrow S_w(e^{j\omega}) \quad (3.17)$$

$$\Rightarrow S_w(e^{j\omega}) = \left[(2\pi)\sum_{k=-\infty}^{k=\infty} \frac{1}{4}\delta(\omega + 2\pi k)\right] + \frac{1}{4} \quad (3.18)$$

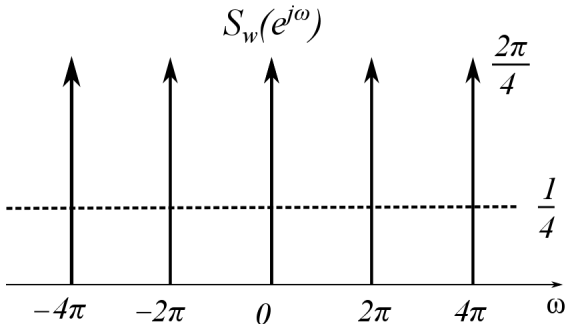


Figure 3.14: PSD of $w[i]$

The white noise level of $\frac{1}{4}$ is because of the random transitions of data and, consequently, sampling impulses at random integer multiplies of the clock, $w[i]$. According to (3.12), in order to find $S_{PD_{out}}$ we need to convolve $S_{f(\Delta\phi)}(e^{j\omega})$ and $S_w(e^{j\omega})$ to find PSD of PD_{out}.

$$\begin{aligned} S_{PD_{out}}(e^{j\omega}) =& \frac{1}{2\pi}\int_{2\pi}(\sum_{k=-\infty}^{k=\infty}\frac{2\pi}{4}\delta(\lambda+2\pi k)+\frac{1}{4})S_{f(\Delta\phi)}(e^{j(\omega-\lambda)})d\lambda \\ &= \frac{1}{4}S_{f(\Delta\phi)}(e^{j\omega})+\frac{1}{2\pi}\int_{2\pi}\frac{1}{4}S_{f(\Delta\phi)}(e^{j\lambda})d\lambda \end{aligned} \tag{3.19}$$

As Figure 3.15 shows, the original spectrum is scaled by $\frac{1}{4}$, which can be viewed as a PD gain of $\sqrt{\frac{1}{4}}=\frac{1}{2}$. In addition there is a white noise, which is the second term in (3.19), and its energy is one fourth of the mean squared value of $f(\Delta\phi)$. This is the noise level in $S_w(e^{j\omega})$, due to data random transitions. Therefore, we name it Random Transition Noise (RTN). This process can be modeled by the block diagram of Figure 3.16.

In [31] Duttweiler investigates the effect of random transitions on the output jitter, assuming large static phase errors. This assumption no longer holds in modern systems that employ Type-II loops. Roza has also studied the effect of random transitions on the output jitter [32]. Interestingly, the noise he calculates due to random transitions is similar to what we have found of a CDR, but he ignores the VCO noise. Moreover, his time domain approach does not provide much intuition, and makes it difficult to calculate the effect of the other noise sources, especially in a non-linear loop.

3.4.1 Alexander Phase Detector

This PD samples $sgn(\Delta\phi)$, nd holds the constant at its output for 1 UI ($T=\frac{1}{f_{clk}}$). This process is a conversion of discrete time samples into their continuous time counterparts and passing them through a Zero-Order Hold, as is shown in Figure 3.17. If we view this operation in frequency domain, it is a multiplication by $H(f)=\frac{\sin(\pi\frac{f}{f_s})}{\pi\frac{f}{f_s}}e^{-j\pi\frac{f}{f_s}}$.

For the case of the Alexander PD $f(\Delta\phi)=sgn(\Delta\phi)=\pm1$. Let m denote the probability

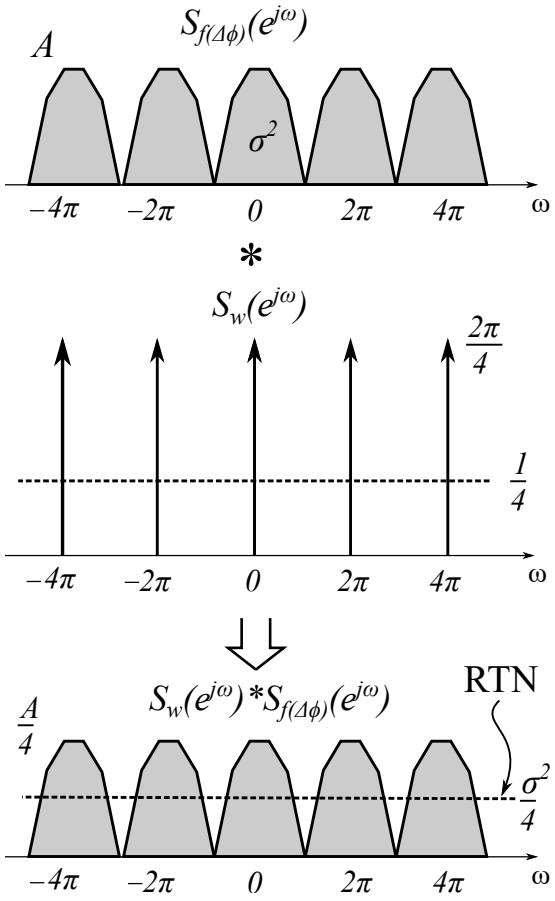


Figure 3.15: Convolving S_w and $S_{f(\Delta\phi)}$

that $sgn(\Delta\phi) = 1$. We want to find the output spectrum. Thus, we need to know the PSD of $sgn(\Delta\phi)$. We start with its auto-correlation function.

$$R_{sgn(\Delta\phi)}[n] = E\left(sgn(\Delta\phi)_{n+k} sgn(\Delta\phi)_k\right) \tag{3.20}$$

$$R_{sgn(\Delta\phi)}[n=0] = (+1)^2 m + (-1)^2(1-m) = 1 \tag{3.21}$$

As was mentioned earlier ISI shifts zero crossings from their ideal positions. Therefore, previous bits will affect the present phase error, which might correlate different $\Delta\phi[i]$s. The Probability Density Function (PDF) of ISI in many applications is modeled by Dual-Dirac function [33–35], although higher number of impulses is possible. In the analysis presented here we focus on Dual-Dirac distribution, but this analysis can be extended for other cases as well.

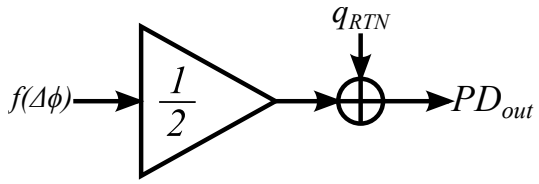


Figure 3.16: Random sampling linear model

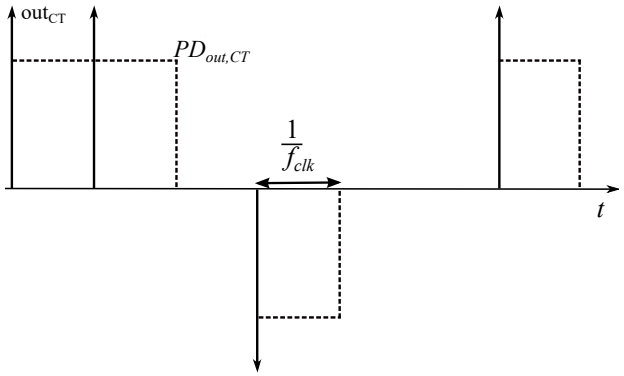


Figure 3.17: Continuous Time output of Alexander PD. Solid: Continuous Time impulses. Dashed: Output of a Zero-Order Hold

The Dual-Dirac distribution suggests that there are two possible situations that determine ISI. Analui in [23] states that for most practical cases one[3] of the previous bits has the most significant effect on this deviation of zero crossing from its ideal position at each transition. This most significant bit is normally the bit before the transition. It means that, for example, depending on whether we have "101" or "001" two possible values for zero crossing will be observed, and the effect of other bits may be neglected.

According to this argument and the independence of the bits for that determine the ISI at every comparison, we claim that ISI does not create correlation between phase errors[4]. In addition, the loop inside which the phase detector is employed is very narrowband and inevitably slow: 100∼1000 times slower than the incoming data. This implies that a single phase error will have a minor effect[5] on the future loop output samples and consequently

[3]This number is bigger than one for more than two impulses

[4]Phase errors are slightly correlated when multiple impulses are present in the ISI PDF

[5]Less than 1%

on subsequent phase errors. As a result, we neglect the correlation between different phase error samples. Thus,

$$\begin{aligned} R_{sgn(\Delta\phi)}[n \neq 0] &= E\left(sgn(\Delta\phi)_{n+k} sgn(\Delta\phi)_k\right) \\ &= E\left(\left(sgn(\Delta\phi)_{n+k}\right) \times E\left(sgn(\Delta\phi)_k\right)\right. \\ &= \underbrace{2m(1-m)(1)(-1)}_{1\&-1} + \underbrace{m^2(1)^2}_{\text{Both: } 1} + \underbrace{(1-m)^2(-1)^2}_{\text{Both:}-1} \\ &= (2m-1)^2 \end{aligned} \tag{3.22}$$

Depicted in Figure 3.18, this auto-correlation function can be expressed as $R_{sgn(\Delta\phi)}[n] = (2m-1)^2 + (1-(2m-1)^2)\delta[n]$, and, hence, the PSD as shown in Figure 3.19 is:

$$S_{sgn(\Delta\phi)}(e^{j\omega}) = [2\pi(2m-1)^2 \sum_{k=-\infty}^{k=\infty} \delta(\omega + 2\pi k)] + 4m - 4m^2 \tag{3.23}$$

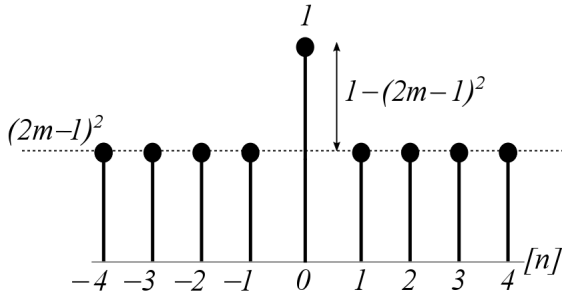


Figure 3.18: The auto-correlation of $sgn(\Delta\phi)$

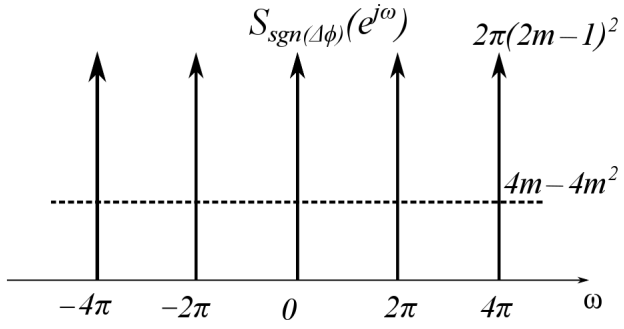


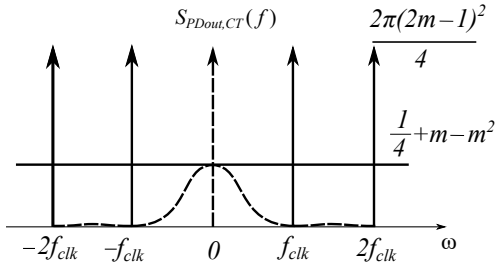
Figure 3.19: The power spectral density of $sgn(\Delta\phi)$

Using (3.19) we can find $S_{PD_{out}}$.

$$\begin{aligned} S_{PD_{out}}(e^{j\omega}) =& \frac{1}{2\pi}\int_{2\pi}(\sum_{k=-\infty}^{k=\infty}\frac{2\pi}{4}\delta(\lambda+2\pi k)+\frac{1}{4})S_{f(\Delta\phi)}(e^{j(\omega-\lambda)})d\lambda \\ &= \frac{1}{4}S_{f(\Delta\phi)}(e^{j\omega})+\frac{1}{2\pi}\int_{2\pi}\frac{1}{4}S_{f(\Delta\phi)}(e^{j\lambda})d\lambda \\ &= [\frac{2\pi}{4}(2m-1)^2\sum_{k=-\infty}^{k=\infty}\delta(\omega+2\pi k)]+m-m^2+\frac{1}{4} \end{aligned} \tag{3.24}$$

This waveform is converted into continuous-time impulses and passes through the described Zero-Order Hold. The resultant spectrum of this process is shown in Figure 3.20.

It could be readily seen if $m \neq \frac{1}{2}$ there is a "tone" at DC. We will discuss in detail a complete loop in Chapter 4 where the output of PD is fed as the error signal to a Type-II feedback loop. This DC component of phase error will be suppressed to zero by feedback loop, and only the colored noise remains. This implies that when lock is achieved, (from Figure 3.20) m must be $\frac{1}{2}$ and $E\left(sgn(\Delta\phi)\right) = 0$.



Solid: Before passing through Zero-Order Hold, dashed: After Zero-Order Hold

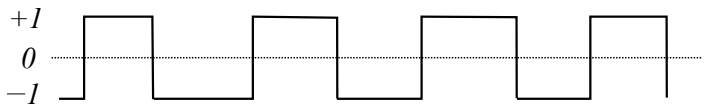
Figure 3.20: The continuous time spectrum of Alexander PD output.

3.4.2 Hogge Phase Detector

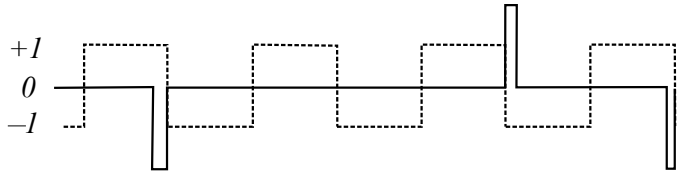
A typical waveform at the Hogge PD output is shown in Figure 3.21 (a). We can model it with a square wave to which pulses are added whose widths are proportional to $w[i]\Delta\phi[i]$ (Figure 3.21 (b)). If the phase difference is small, the width modulated pulses will have a very short duration and could be approximated with impulses with the same area, as is

shown in Figure 3.21 (c). We therefore decompose the output into a 50% waveform and a train of amplitude modulated impulses.

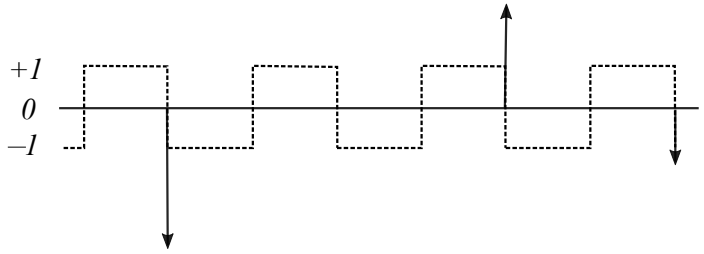
$$S_{PD}(f) = S_{PD_{out,CT}}(f) + S_{SquareWave}(f) \tag{3.25}$$



(a) Hogge PD output



(b) Decomposition into 50% duty cycle waveform (dashed) and PWM waveform (solid)



(c) Approximation by 50% duty cycle waveform (dashed) and train of impulses (solid)

Figure 3.21: Hogge PD continuous time waveform

$PD_{out,CT}$ is obtained when discrete phase errors $f(\Delta\phi_k)$ are converted to their continuous

counterparts. We need the auto-correlation of $f(\Delta\phi)$ to find its spectrum.

$$R_{f(\Delta\phi)}[n] = E\bigg(f(\Delta\phi_k)f(\Delta\phi_{n+k})\bigg) \tag{3.26}$$

If $n = 0$:

$$\begin{aligned} R_{f(\Delta\phi)}[n=0] &= E\bigg(f(\Delta\phi_k)^2\bigg) \\ &= E\bigg((\frac{\Delta\phi_k}{2\pi})^2\bigg) \\ &= \sigma^2_{\Delta\phi} \cdot \frac{1}{(2\pi)^2} \end{aligned} \tag{3.27}$$

If $n \neq 0$:

$$\begin{aligned} R_{f(\Delta\phi)}[n\neq 0] &= E(f(\Delta\phi_k)f(\Delta\phi_{n+k})) \\ &= E(f(\Delta\phi_k)) \times E(f(\Delta\phi_{n+k})) \\ &= m^2_{\Delta\phi} \cdot \frac{1}{(2\pi)^2} \end{aligned} \tag{3.28}$$

$m_{\Delta\phi}$ and $\sigma_{\Delta\phi}$ denote mean and mean squared value of $\Delta\phi$, respectively. Using the similar argument to the case of Alexander PD we can deduce that in the locked condition in a Type-II PLL, $m_{\Delta\phi}$ must be 0. Thus,

$$R_{f(\Delta\phi)} = \frac{\sigma^2_{\Delta\phi}}{(2\pi)^2}\delta[n] \leftrightarrow S_{f(\Delta\phi)}(e^{j\omega}) = \frac{\sigma^2_{\Delta\phi}}{(2\pi)^2} \tag{3.29}$$

We use (3.19) to find $S_{PD_{out}}(e^{j\omega})$:

$$S_{PD_{out}}(e^{j\omega}) = \frac{\sigma^2_{\Delta\phi}}{2(2\pi)^2} \tag{3.30}$$

So if we replace discrete time impulses with continuous time ones:

$$S_{PD_{out,CT}}(f) = \frac{\sigma^2_{\Delta\phi}}{2(2\pi)^2}, \tag{3.31}$$

and the complete spectrum of the output waveform will be obtained by adding the spectrum of a square waveform to (3.31):

$$S_{PD}(f) = \frac{\sigma^2_{\Delta\phi}}{2(2\pi)^2} + \sum_{k=-\infty}^{k=\infty}(\frac{4}{(2k-1)\pi})^2\delta(f-(2k-1)f_{clk}) \tag{3.32}$$

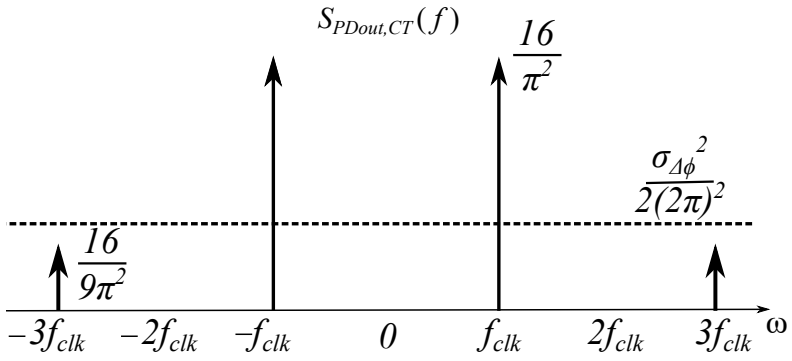


Figure 3.22: The continuous time spectrum of Hogge PD output

It could be readily seen that, unlike Alexander PD, the output of Hogge PD contains harmonics at the clock frequency (Figure 3.22). We will discuss in Chapter 3 that this will lead to Duty Cycle Distortion (DSD).

CHAPTER 4

Clock Data Recovery Loop Analysis

PLLs and CDR loops are very similar to each other. A great deal of our knowledge of PLLs could be applied to CDRs as well. This chapter includes as yet unpublished works of Hao Xu, a fellow graduate student, and includes a simple and accurate analysis of these loops. We focus our analysis on the case of PLLs because of their generality.

4.1 Phase Domain Modeling of PLLs

If we take voltage as our input to PLLs and CDRs they include non-linear blocks, such as phase detectors and Voltage Controlled Oscillators (VCOs). However, these blocks can be treated linearly if we accept phase as our input. [36] presents an accurate linear model for these loops.

Phase detector, as complicated as it might be, only extracts the phase difference between its inputs. Therefore, it amounts to a simple subtraction in phase domain. A Charge Pump (CP) usually follows the phase detector. It converts the phase domain information into current, but this conversion has no importance as far as loop behavior is concerned. Loop filter operates on the charge-pump generated current. The current passes through a filter and produces a voltage. Again, if we neglect the dimension change, it only filters the input.

Charge pump is replaced by a digital filter in All Digital PLLs (ADPLLs). This modification does not cause any problems since digital filter's input is voltage which is provided by the phase detector, and a charge pump is not necessary as far as dimension transformation is concerned. Charge Pump functionality is also built into the loop filter by including a digital

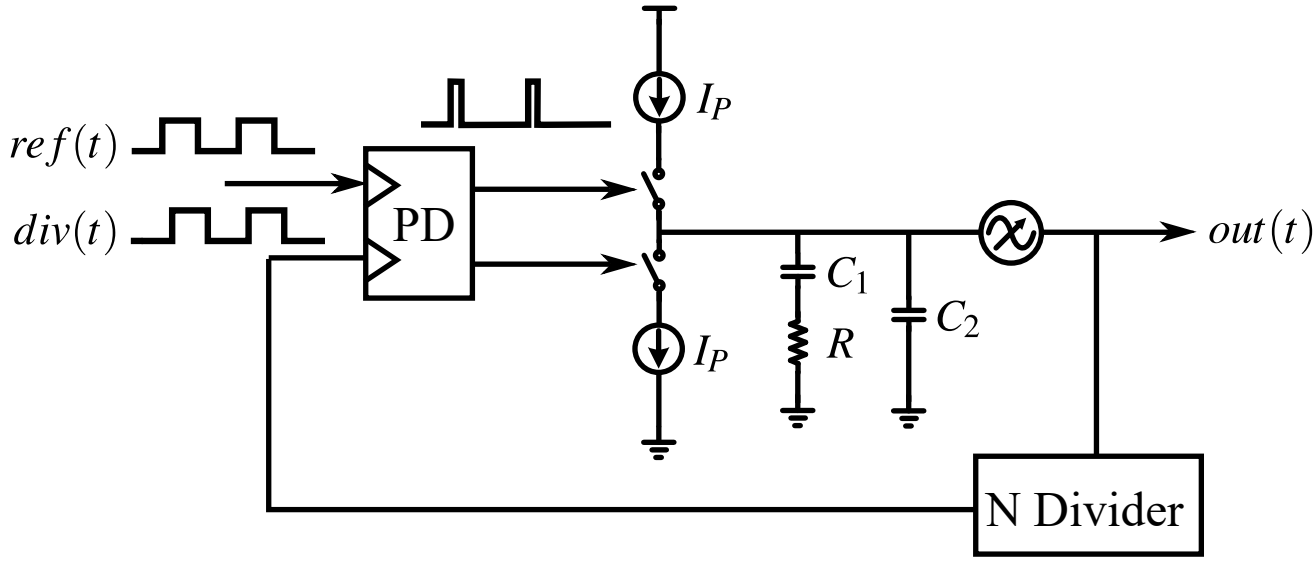


Figure 4.1: A charge-pump PLL

integration $\frac{1}{1-z^{-1}}$.

The final block is the VCO, an oscillator whose output frequency is controlled by a control voltage taken as the loop filter output in this system. The output frequency of VCO is modeled as linearly dependent on control voltage:

$$\omega = k_{VCO} V_{cntrl} + \omega_0 \tag{4.1}$$

K_{VCO} unit is $\frac{\text{rad}}{\text{s.Volt}}$ and ω_0 is the VCO's free-running frequency.

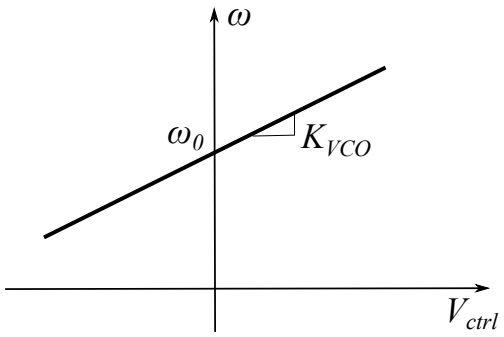


Figure 4.2: VCO frequency versus control voltage

Phase is the integration of frequency, hence:

$$\phi(t) = \int_{-\infty}^{t} k_{VCO} V_{cntrl}(\tau) d\tau \tag{4.2}$$

Each of described blocks is LTI and, therefore, can be expressed in s-domain.

The loop for a CDR is usually a Type-II loop which brings about a zero static phase error.

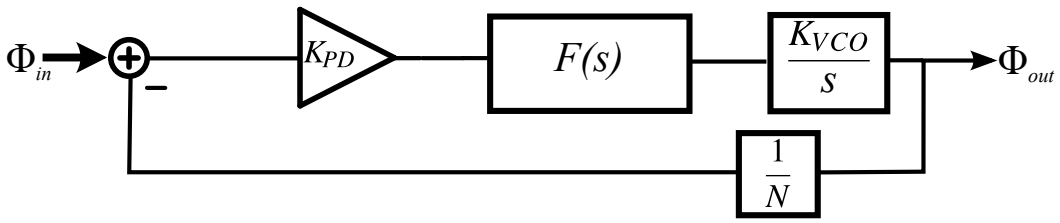


Figure 4.3: S-domain representation of a PLL

The loop type is dictated by the number of poles at $s = 0$ in the loop gain. In this system a pole at zero is contributed by the integration in VCO and another from the integrator in the loop filter. It is readily seen that such loop will not be stable unless it contains a Left Half Plane (LHP) zero to provide sufficient phase margin. This zero is implemented inside the loop filter. Furthermore when the Hogge phase detector is used, a spectral component is present at the clock frequency in the phase detector output. This component appears in the VCO control voltage and, thus, modulates the VCO frequency. As we know, if a tone at f_0 gets Frequency Modulated with a carrier f_c, tones at $f_c + f_0$ and $f_c - f_0$ will be generated. In this case, $f_c = f_0 = f_{clk}$, and the generated tones will be at DC and $2f_{clk}$. Having a tone at $2f_{clk}$ implies that the VCO output does not have a 50% duty cycle. DCD[1] is undesirable and needs to be controlled. Hence, the PD component at f_{clk} needs to be suppressed by the loop filter which might require an additional pole (at ω_{p2}) in the loop filter. (4.3) is a general case for loop filter transfer function. If the loop filter does not have the second pole, as is the case when using the Alexander phase detector, we can assume $\omega_{p2} \to \infty$ in the results of the following analysis.

$$F(s) = (\frac{\alpha}{s} + \beta)(\frac{1}{\frac{s}{\omega_{p2}} + 1}) \tag{4.3}$$

[1]Duty Cycle Distortion

4.2 Linearization of Bang Bang Phase Detector (BBPD)

Bang bang phase detection is a non-linear process. The output is simply the sign of the phase error irrespective of the error magnitude. A lot of prior works have tried to linearize the characteristics of this block so we can apply our knowledge of linear systems to it. [8] and [9] have proposed a linearization method for small phase errors. [10] relies on simulation for calculating the quantization noise spectrum and does not give any analytical proof for its spectrum. Da Dalt in [37] performs the linearization with an analysis based on Markov chains. Park and Kim [13] assume a constant phase difference and ignore the deterministic jitter when calculating the gain. They assume a normal distribution for the input phase error, which is a good approximation for the case of frequency synthesizer. However, deterministic jitter in a link with random data causes the phase error distribution to deviate from normal, and makes it close to Dual-Dirac distribution [33, 34]. Here we calculate the gain using this corrected PDF for the incoming phase error and also clearly define the quantization noise and explain how it cannot be correlated to the input, using a signal space representation. In addition, [13] only relates the BBPD gain to the noise level at its input, but does not give an expression for that noise. However, we analytically calculate this noise level and, subsequently, we are able to obtain an accurate closed form expression for the output jitter.

The output of a non-linear system y and its input x are shown in signal space domain in Figure 4.4. If we extract the input correlated component of the output, $c \cdot x$, the residue error q will be orthogonal to the input. The scaling factor of correlated part, c, will be considered as gain and the residue error acts as a noise source, which is uncorrelated to the input because of the orthogonality.

$$y = c.x + q \tag{4.4}$$

So as to calculate the gain we note that:

$$E(yx) = c.E(x^2) + E(qx)$$

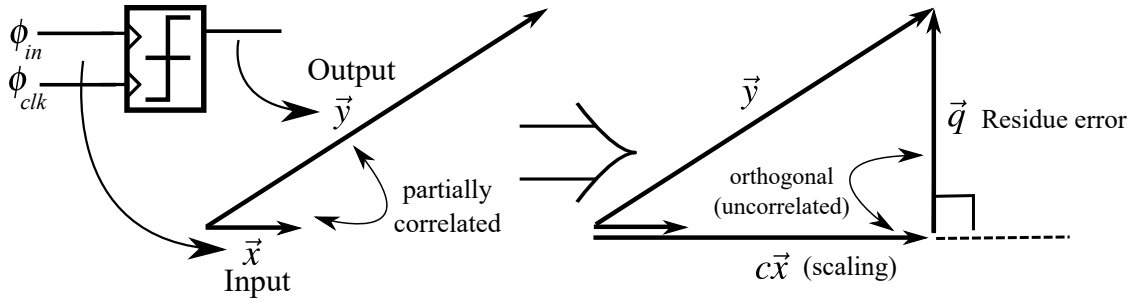


Figure 4.4: Signal space decomposition of BBPD output

q and x are orthogonal which means $E(qx) = 0$ and implies a zero correlation between them. Therefore,

$$c = \frac{E(yx)}{E(x^2)} \tag{4.5}$$

Also we can find the energy of q using Pythagoras' theorem in signal space.

$$E(q^2) = E(y^2) - c^2 E(x^2) \tag{4.6}$$

The incoming jitter can be decomposed into a deterministic component due to ISI and a random component. We know the random component is produced by various independent sources. Although central limit theorem only applies to linear superposition of independent noise sources which is violated by Bang-Bang PD, Hao's result shows that it still remains a good approximation. Thus, the random component has a Gaussian distribution with standard deviation σ_r, where the subscript r stands for "random". We also assume a Dual-Dirac distribution at $+d$ and $-d$ for the deterministic jitter, as Figure 4.5 shows. The gain could be calcu-

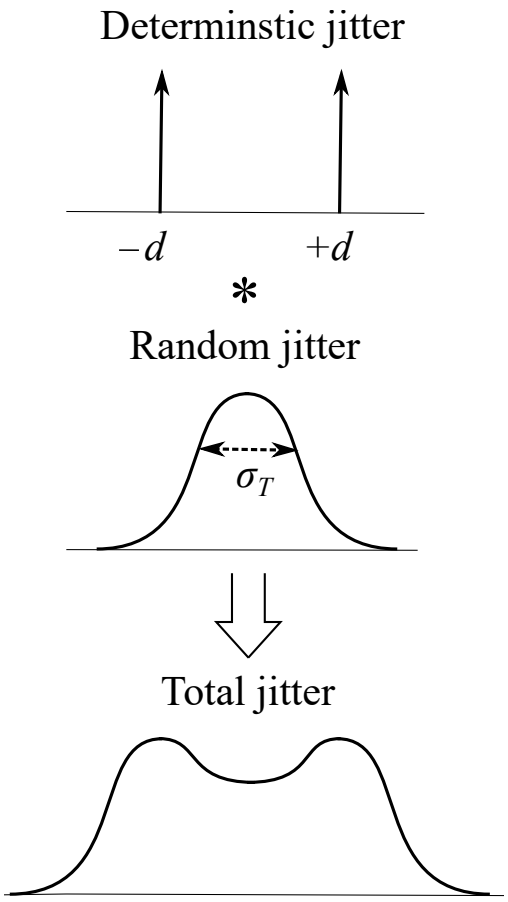


Figure 4.5: Total jitter at the phase detector input

lated as follow.

$$K_{BPD} = \frac{E(sgn(x) \times x)}{E(x^2)}$$

$$K_{BPD} = \frac{\sqrt{\frac{2}{\pi}}\sigma_r e^{-\frac{d^2}{2\sigma_r^2}} + d(Q_{func}(-\frac{d}{\sigma_r}) - Q_{func}(\frac{d}{\sigma_r}))}{\sigma_r^2 + d^2} \tag{4.7}$$

where $Q_{func}(x)$ determines the probability that a normal variable is greater than x.
If we define $\sigma = \sqrt{\sigma_r^2 + d^2}$, when deterministic jitter is very small (i.e. $d \ll \sigma_r$), this expression simplifies to

$$K_{BPD} = \sqrt{\frac{2}{\pi}}\frac{1}{\sigma} \tag{4.8}$$

which has been found for BBPD gain in previous works [8–10,12,13] and is very close to [37].

If $d < 2\sigma_r$, which is the case for most applications with effective equalization, the calculated gain from (4.7) matches the normal approximation (4.8) within 12%. Even when $d > 2\sigma_r$, the difference between (4.7) and (4.8) does not exceed 20%. The error between these two expressions is plotted in Figure.4.6.

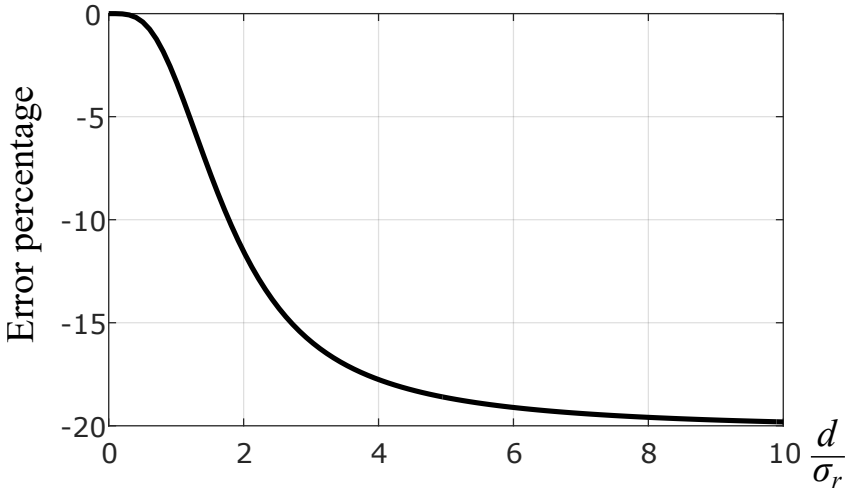


Figure 4.6: The error between the gain with a normal approximation in (4.8) and the exact gain in (4.7)

The additive noise, which we label "quantization noise" is quantified by (4.6).

$$\sigma_q^2 = E(q^2) = 1 - K_{BPD}^2 \sigma^2 \tag{4.9}$$

where σ^2 is the total jitter mean squared value.

Unlike gain, the quantization noise changes dramatically with deterministic jitter. If the deterministic jitter is small ($d < \sigma_r$), we can continue to use the approximation of a normal distribution and write

$$\sigma_q^2 = E(q^2) \approx 1 - \frac{2}{\pi} \tag{4.10}$$

The error between the exact value and the approximation is within 10%. This function is plotted in Figure 4.7.

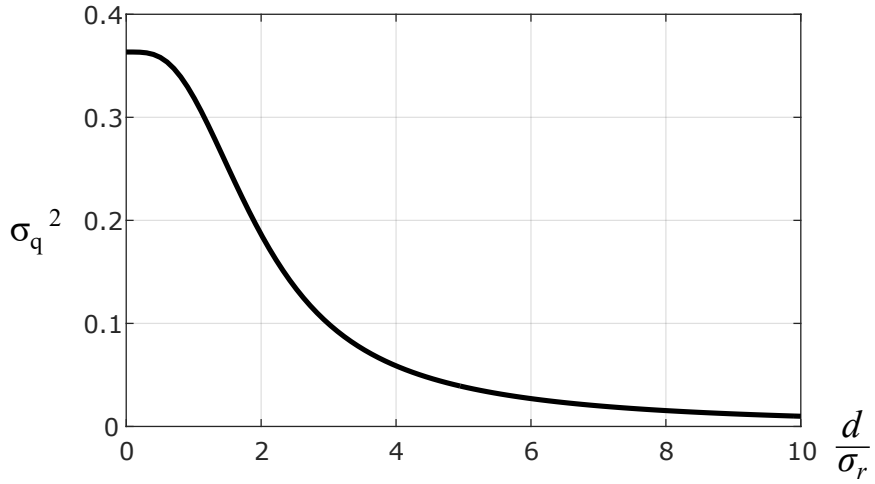


Figure 4.7: σ_q^2 for different deterministic jitters

We assume a white spectrum for the input noise, but we need to prove q also has a white spectrum. Because q by definition is uncorrelated with the random input phase at the PD output, we claim the values it takes are independent from one phase comparison to another. Each of these values is held for a period T_{ref}, the phase sampling period, until the next comparison is done. Therefore, its auto-correlation decreases linearly when we shift it, and goes to zero if it is shifted more than T_{ref}. We can express the auto-correlation function to

be

$$R_q(\tau) = \sigma_q^2 \Lambda(\frac{\tau}{T_{ref}}) \tag{4.11}$$

In this equation $\Lambda(t)$ denotes the triangular function.

$$\Lambda(t) = \begin{cases} 1 - |t| & |t| < 1 \\ 0 & \text{o.w} \end{cases} \tag{4.12}$$

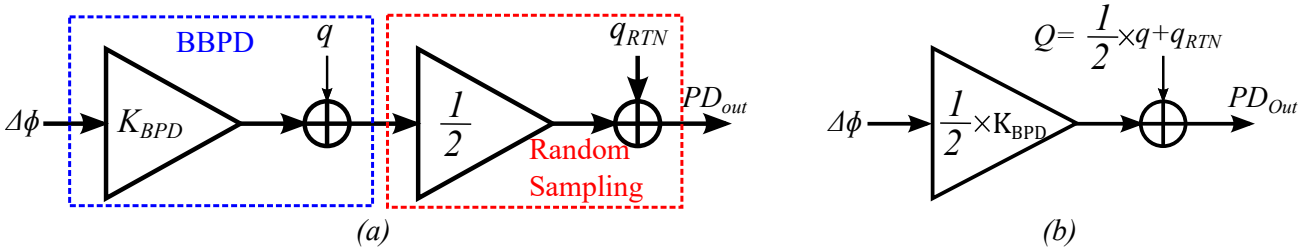
This auto-correlation function corresponds to the following PSD.

$$S_q(f) = 2T_{ref}\sigma_q^2 Sinc^2(fT_{ref}) \tag{4.13}$$

This noise source only sees low pass transfer functions in the loop, whose upper cutoff is equal to the loop unity gain frequency f_u. Since these loops are designed to have a very small bandwidth compared to their reference frequency, f_{ref}, $S_q(f)$ can be approximated by a constant level over the frequency interval $(0, f_{ref})$ and, therefore, considered to be white.

The mentioned gain is only associated to PD behavior. According to our discussion in the previous chapter, if this phase detector is used for a CDR, the equivalent system may be modeled as Figure 4.8 (a). If we associate all noise sources to the output node we will end up with the block diagram in Fig 4.8 (b)

If a Hogge detector is used $q = 0$ and $K_{BPD} = \frac{1}{2\pi}$.



(a) BBPD followed by random sampling, (b) Integration of amplifications and noise sources

Figure 4.8: CDR BBPD modeling

It was proved in (3.19) that the mean squared value of RT noise is one fourth of the PD output mean squared value. In the case of BBPD :

$$\sigma_{q_{RTN}}^2 = \frac{1}{4}\sigma_{sgn(\Delta\phi)}^2 = \frac{1}{4} \tag{4.14}$$

From Figure 4.8 (b) we can calculate the mean squared value of Q, the effective noise for a BBPD to be:

$$\sigma_Q^2 = (\frac{1}{2})^2 \times \sigma_q^2 + \sigma_{q_{RTN}}^2 \tag{4.15}$$

For $d < \sigma_r$, this simplifies to

$$\sigma_Q^2 = (\frac{1}{2})^2 (1 - \frac{2}{\pi}) + \frac{1}{4} = \frac{\pi - 1}{2\pi} \tag{4.16}$$

In the Hogge PD the effective noise depends on the mean squared value of the incoming phase noise, according to (3.29)

$$\sigma_Q^2 = \frac{1}{4} \cdot \frac{\sigma_{\Delta\phi}^2}{(2\pi)^2} \tag{4.17}$$

4.3 s-Domain Loop Model and Noise Sources

Figure 4.9 shows the loop in s-domain. The noise source V_n intends to model the VCO phase noise by a noise voltage. If the noise parameter of VCO is K_w, the VCO phase noise equals:

$$S_{\phi VCO} = \frac{K_w}{f^2} \tag{4.18}$$

The phase noise produced by V_n equals $S_{V_n} \left| \frac{K_{VCO}}{2\pi f} \right|^2$. By equating this to (4.18).

$$S_{V_n} = (2\pi)^2 \frac{K_w}{K_{VCO}^2} \tag{4.19}$$

This value is constant over frequency suggesting a white spectrum for V_n.
We let $T(s)$ denote the open loop gain.

$$T(s) = K_{PD} F(s) K_{VCO} \frac{1}{N} \tag{4.20}$$

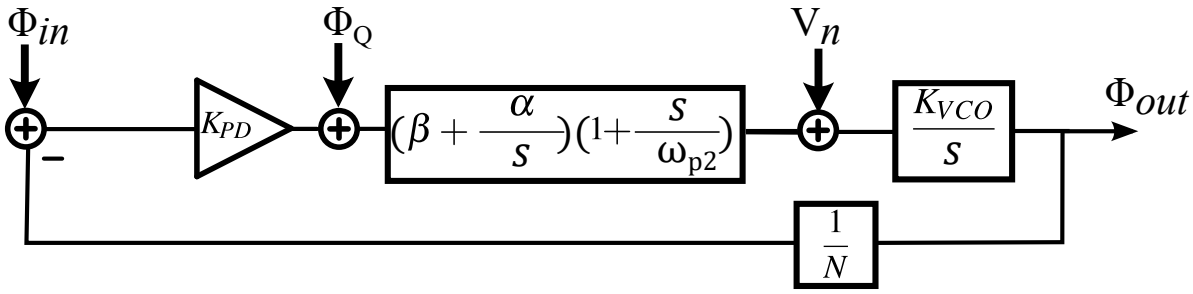


Figure 4.9: CDR Loop in s-domain

In most wireline applications jitter peaking should not exceed certain values which requires adequate phase margin. Thus, well separated poles and zeros are needed. The goal of the second pole, ω_{p2} is to attenuate clock feed through at PD output, Hence, it is located at much higher frequencies than ω_u so as not to erode phase margin. As a result of the fact that phase margin is controlled by the zero, $\omega_z = -\frac{\alpha}{\beta}$, position, it needs to be low enough to provide desired phase margin. Typically it is placed three or four times lower than the unity gain frequency and the second pole is located three or four times higher than ω_u. Given these conditions are met we can calculate ω_u.

$$\omega_u = 2\pi f_u = K_{PD}\beta K_{VCO}\frac{1}{N} \tag{4.21}$$

With these definitions and approximations we can reformulate $T(s)$.

$$T(s) = \frac{\omega_u}{s}.\frac{1+\frac{\omega_z}{s}}{1+\frac{s}{\omega_{p2}}} \tag{4.22}$$

4.4 Loop Parameters Calculation

We have so far set up a loop and defined all of its parts, but in order to calculate phase noise at the output we need numerical values for the loop parameters.

It is discussed in section 4.2 that the effective quantization noise for both phase detector types as well as the gain for BBPD depend on the noise at phase detectors input, σ_{ϕ_e}. In

this section we use equivalent noise bandwidth, NBW, to derive a closed form expression for σ_{ϕ_e}.

The noise at PD's input can be decomposed into three components due to three noise sources in the system: Input, VCO, and PD.

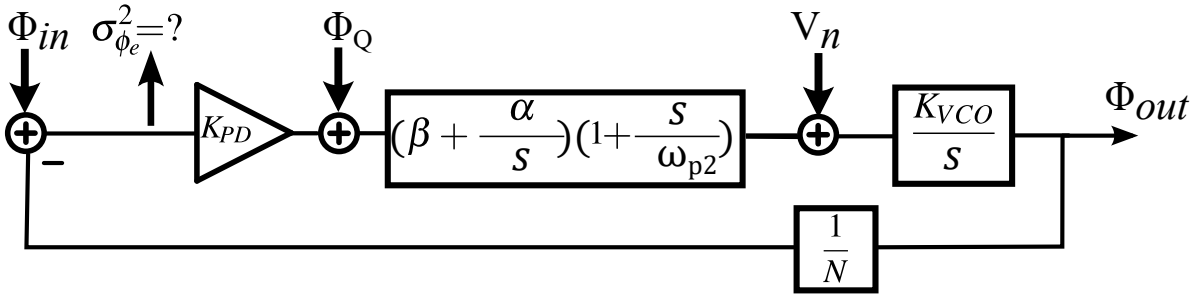


Figure 4.10: BBPD Input phase noise calculation

$$\Phi_e = \Phi_{In}\frac{1}{1+T} + \Phi_Q\frac{1}{K_{PD}}\frac{T}{1+T} + V_n\frac{K_{VCO}}{sN}\frac{1}{1+T} \tag{4.23}$$

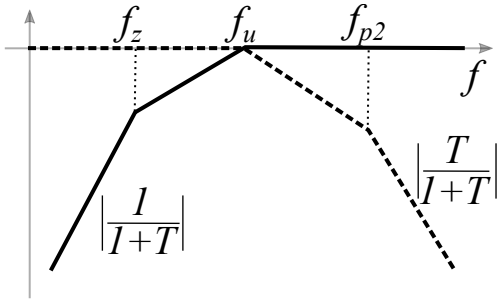
All of these noise sources are uncorrelated and we can find the PSD of ϕ_e by the following formula.

$$S_{\phi_e} = S_{\phi_{In}}|\frac{1}{1+T}|^2 + S_{\phi_Q}\frac{1}{K_{PD}^2}|\frac{T}{1+T}|^2 + S_{V_n}\frac{K_{VCO}^2}{|2\pi fN|^2}|\frac{1}{1+T}|^2 \tag{4.24}$$

What we are interested in is the mean squared value of ϕ_e. It could be found by integration of the PSD of ϕ_e.

The first term captures the effect of input phase noise at PD input. The transfer function $\frac{1}{1+T}$ exhibits a high pass behavior (Figure 4.11) with low frequency cut off of f_u. Since this frequency is much lower than f_{clk} almost all of the input jitter appears in σ_{ϕ_e}.

$$\sigma_{\phi_e}^2(input) = \int_0^\infty S_{\phi_{In}}|\frac{1}{1+T}|^2 df \approx \frac{\sigma_{In}^2}{\frac{f_{clk}}{2}}.\frac{f_{clk}}{2} = \sigma_{In}^2 \tag{4.25}$$



Dashed: $\frac{T}{1+T}$, solid: $\frac{1}{1+T}$

Figure 4.11: High pass and low pass functions in a loop

The PD effective noise is filtered by $\frac{T}{1+T}$, which is the low pass filter shown in Figure 4.11, and we are interested in

$$\sigma^2_{\phi_e}(Q) = \int_0^\infty S_{\phi_Q} \frac{1}{K^2_{PD}} |\frac{T}{1+T}|^2 df \tag{4.26}$$

ϕ_Q has a white spectrum for our purposes, as was mentioned earlier. Therefore we can use NBW. We know if we have a second order transfer function $H(s)$, we can associate an effective noise bandwidth to it.

$$H(s) = \frac{1}{\frac{s^2}{\omega_0^2} + \frac{s}{\omega_0 Q} + 1} \Rightarrow NBW_{LP} = \frac{\omega_0}{4Q} = \frac{\pi}{2}\frac{f_0}{Q}(Hz) \tag{4.27}$$

Assuming widely separated poles and zeros and $\omega_{p2} \ll \omega_u$

$$\frac{T}{1+T} \approx \frac{1}{(1+\frac{s}{\omega_u})(1+\frac{s}{\omega_{p2}})} \Rightarrow NBW_{LP} \approx \frac{\omega_u}{4} = \frac{\pi}{2} f_u \tag{4.28}$$

Therefore the result of the integral in (4.26) will be :

$$\int_0^\infty S_{\phi_Q} \frac{1}{K^2_{PD}} |\frac{T}{1+T}|^2 df \approx \frac{\sigma^2_Q}{\frac{f_{clk}}{2}} \frac{1}{K^2_{PD}} . NBW_{LP} = \sigma^2_Q \frac{\pi}{K^2_{PD}} \frac{f_u}{f_{clk}} \tag{4.29}$$

The third term in (4.24) is the effect of VCO noise which experiences a bandpass function.

$$\sigma^2_{\phi_e}(VCO) = \int_0^\infty S_{V_n} \frac{K^2_{VCO}}{|sN|^2} |\frac{1}{1+T}|^2 df \tag{4.30}$$

We can again use NBW, because V_n has a white spectrum. A standard formulation of a second order bandpass filter and its equivalent noise bandwidth is presented by (4.31).

$$H(s) = \frac{\frac{s}{\omega_0 Q}}{\frac{s^2}{\omega_0^2} + \frac{s}{\omega_0 Q} + 1} \Rightarrow NBW_{BP} = \frac{\omega_0}{4} Q = \frac{\pi}{2} f_0 Q(Hz) \tag{4.31}$$

We can rearrange the transfer function seen by V_n to match the standard format, assuming a well designed loop.

$$\frac{1}{s} \cdot \frac{1}{1+T} \approx \frac{1}{\omega_u} \frac{\frac{s}{\omega_z}}{\frac{s^2}{\omega_u \omega_z} + \frac{s}{\omega_z} + 1} \Rightarrow NBW_{BP} = \frac{\pi}{2} f_u \tag{4.32}$$

By employing this equation we can find $\sigma^2_{\phi_e}(VCO)$.

$$\sigma^2_{\phi_e}(VCO) = \int_0^\infty S_{V_n} \frac{K^2_{VCO}}{|sN|^2} |\frac{1}{1+T}|^2 df \approx (2\pi)^2 \frac{K_w}{K^2_{VCO}} \frac{K^2_{VCO}}{N^2\omega_u^2} NBW_{BP} = \frac{K_w}{N^2 f_u} \frac{\pi}{2} \tag{4.33}$$

(4.25), 4.26, and 4.33 express the different components of σ_{ϕ_e}. Putting all together we can write:

$$\sigma^2_{\phi_e} = \sigma^2_{\phi_e}(input) + \sigma^2_{\phi_e}(Q) + \sigma^2_{\phi_e}(VCO) \tag{4.34}$$

4.4.1 Bang Bang Phase Detector Gain and Quantization Noise

Consider the case of low deterministic jitter for now. (4.8) expresses the relation between the BBPD gain and its input phase noise as

$$K_{BPD} = \sqrt{\frac{2}{\pi}} \frac{1}{\sigma_{\phi_e}}$$

We need to pay attention that this is the gain of BBPD due to bang-bang action, but the effective gain seen by the loop is $K_{PD} = \frac{1}{2} K_{BPD}$, as could be seen in Figure 4.8, because of the random sampling. By plugging in (4.8) into 4.34 we will get:

$$\frac{2}{\pi} \frac{1}{K^2_{BPD}} = \sigma^2_{\Phi_{In}} + (\frac{\pi - 1}{2\pi}) \frac{\beta K_{VCO} T_{clk}}{2N K_{PD}} + \frac{\pi^2 K_W}{N\beta K_{PD} K_{VCO}} \tag{4.35}$$

We can solve this quadratic equation to find K_{BPD} in terms of loop parameter.

$$\begin{gathered} \Rightarrow K_{BPD} = -\frac{\sigma^2_{\Phi_0}}{2\sigma^2_{\Phi_{In}}} + \sqrt{\left(\frac{\sigma^2_{\Phi_0}}{2\sigma^2_{\Phi_{In}}}\right)^2 + \left(\frac{2}{\pi}\right) \cdot \frac{1}{\sigma^2_{\Phi_{In}}}} \\ \text{where we defined } \sigma^2_{\Phi_0} = \frac{2\pi^2 K_W}{N\beta K_{VCO}} + \frac{\pi - 1}{2\pi} \frac{\beta K_{VCO} T_{clk}}{N} \end{gathered} \tag{4.36}$$

$\sigma^2_{\Phi_0}$ is called the Intrinsic Noise Power.

For a CDR the input jitter is extremely high. Thus $\sigma^2_{\Phi_0} \ll \sigma^2_{In}$. This assumption simplifies the BBPD gain to

$$K_{BPD} \approx \sqrt{\frac{2}{\pi}} \frac{1}{\sigma_{\phi_{In}}} \tag{4.37}$$

This equation suggests that the BBPD gain in CDR is set by the input phase noise only. It is worth noticing that the incoming jitter mean squared $\sigma_{\phi_{In}}$ value encompasses the effect of both random and deterministic jitter in this case.

What would happen if the deterministic jitter is not low and our approximations do not hold?

In this case, we can still use 4.34 to calculate σ_{ϕ_e}. Figure 4.7 shows the Band Bang quantization noise decreases with increasing ISI. Since σ_Q and σ_q are related according to 4.15, we can deduce the quantization noise and, therefore, its effect on σ_{ϕ_e} decreases with increase of ISI. Thus, the spectrum of σ_{ϕ_e} will still be dominated by the incoming phase noise. As a result the deterministic and random component at the PD's input will be identical to those for the incoming data, and we can use equation 4.7 and 4.10 to find the Quantization noise and gain.

4.4.2 Hogge Phase Detector Effective Quantization Noise

If a Hogge PD was employed instead of a BBPD, we know the gain was known in advance. However, as we discussed in section 4.2 the RT noise of the Hogge phase detector depends on its input mean squared jitter, $\sigma^2_{\phi_{in}}$, which is dominated by the incoming jitter, according to our discussion. Hence, from here we can find the RTN for Hogge PD using 4.17

$$\sigma^2_Q(Hogge) = \frac{1}{4}\frac{\sigma^2_{\phi_{In}}}{(2\pi)^2} \tag{4.38}$$

4.5 Output Phase Noise Calculation

Now that we know the PD gain, we can write the transfer function from each of the loop noise sources to the output.

$$\Phi_{out} = \Phi_{In} N \frac{T}{1+T} + \Phi_Q \frac{N}{K_{PD}} \frac{T}{1+T} + V_n \frac{K_{VCO}}{s} \frac{1}{1+T} \tag{4.39}$$

Owing to independent sources of these noises we can add their PSDs to get the output phase noise PSD.

$$S_{\phi_{out}} = S_{\phi_{In}} N^2 |\frac{T}{1+T}|^2 + S_{\phi_Q} \frac{N^2}{K^2_{PD}} |\frac{T}{1+T}|^2 + S_{V_n} |\frac{K_{VCO}}{2\pi f} \frac{1}{1+T}|^2 \tag{4.40}$$

In many cases we are interested in the mean squared value of phase noise. Similar approach that we used for calculating σ_{ϕ_e} with equivalent noise bandwidths is taken here.

$$\sigma^2_{\phi_{out}}(input) = \int_0^\infty S_{\phi_{In}} N^2 |\frac{T}{1+T}|^2 df \approx \frac{\sigma^2_{in}}{\frac{f_{clk}}{2}} N^2 NBW_{LP} = \sigma^2_{in} N^2 \pi \frac{f_u}{f_{clk}} \tag{4.41}$$

$$\sigma^2_{\phi_{out}}(Q) = \int_0^\infty S_{\phi_Q} \frac{N^2}{K^2_{PD}} |\frac{T}{1+T}|^2 df \approx \frac{\sigma^2_Q}{\frac{f_{clk}}{2}} \frac{N^2}{K^2_{BPD} T D^2} NBW_{LP} = \sigma^2_Q \frac{N^2}{K^2_{PD}} \pi \frac{f_u}{f_{clk}} \tag{4.42}$$

$$\sigma^2_{\phi_{out}}(VCO) = \int_0^\infty S_{V_n} |\frac{K_{VCO}}{2\pi f} \frac{1}{1+T}|^2 df \approx K^2_{VCO} (2\pi)^2 \frac{K_w}{K^2_{VCO} (2\pi f_u)^2} NBW_{BP} = \frac{K_w}{f_u} \frac{\pi}{2} \tag{4.43}$$

By adding these components together we can find the overall output mean squared phase noise.

$$\sigma^2_{\phi_{out}} = \sigma^2_{in} N^2 \pi \frac{f_u}{f_{clk}} + \sigma^2_Q \frac{N^2}{K^2_{PD}} \pi \frac{f_u}{f_{clk}} + \frac{K_w}{f_u} \frac{\pi}{2} \tag{4.44}$$

This formula could be used for both types of phase detectors. We just need to choose the correct gain and also adjust σ_Q and K_{PD} accordingly.

CHAPTER 5

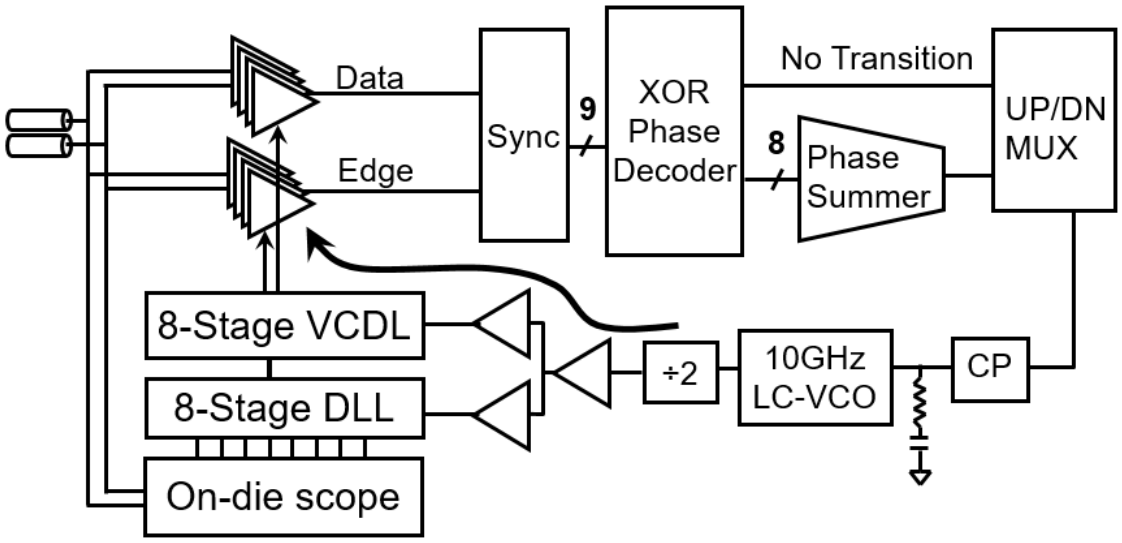
Verification of Theory against Measurement

The analysis introduced in the previous chapter was straightforward and intuitive, but we need to verify its accuracy on a practical circuit. In this chapter we investigate the work of Jaussi *et al.* in [38]. In this work they have presented "A 20 Gb/s Embedded Clock Transceiver in *90nm* CMOS" (Figure5.1).

5.1 System Overview

The incoming data is four 5Gb/s channels. The CDR loop is a subrate architecture [39] using a phase detector that samples the phase error. As we mentioned previously, we need two samples for each decision. Therefore, 40GS/s are needed. In this technology it would be difficult and power hungry to generate high frequency clocks. Instead, a 5GHz clock with 8 phases is used which creates transitions at 40GS/s. These phases are generated with a Voltage Controlled Delay Line (VCDL) which takes its reference from an 8 stage Delay Locked Loop (DLL). The phase detector whose architecture will be studied in detail in the next section makes binary decisions and drives the charge pump.

An LC VCO is preferred because of its lower phase noise compared to the ring oscillator. The clock is generated at 10GHz, and then divided by two to create a 50% duty cycle and overcome duty cycle distortion caused by reference feedthrough.



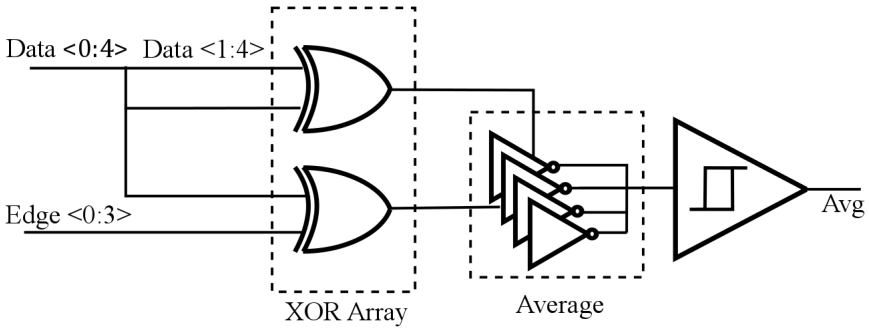
Ref. "A 20Gb/s Embedded Clock Transceiver in *90nm* CMOS", Jaussi *et al.*

Figure 5.1: Receiver architecture of the case study

5.2 Phase Detector Operation

The phase detector uses Early-Late decisions on input transitions. The architecture is very similar to one used in [39]. Figure 5.2 shows the phase detector.

It is always preferred to run the logic at lowest possible frequency in order to save power.



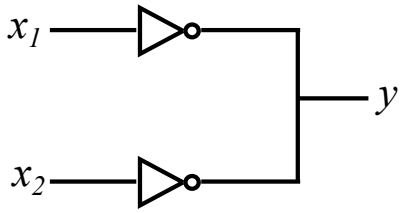
Ref. "A 20Gb/s Embedded Clock Transceiverin 90nm CMOS", Jaussi et al

Figure 5.2: Case study phase detector circuit

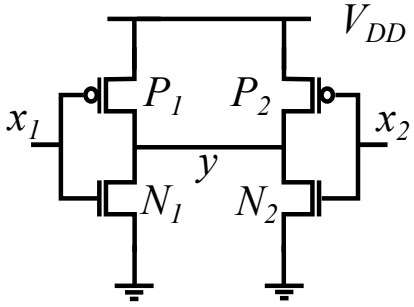
Here the PD produces one output per four incoming bits, determined by the majority. We will discuss this method of decimation shortly.

Per each incoming bit a decision is made by the corresponding XOR, which is identical to an Alexander-type output. If there is a transition in the data waveform (whose probability is 50%) the output is equiprobable $+1$ or -1. But if there is no transition the output is 0. We denote the output with $w[i]sgn(\Delta\phi[i])$ (see Figure 3.12), where before $w[i]$ is defined by, $+1$ if data transitions and 0 otherwise.

In order to understand the function of the average block in Figure 5.2, we focus on the case where the outputs of two identical inverters are tied together (Figure 5.3). If $x_1 = x_2 = 1$, then P_1 and P_2 are on and y is connected to V_{DD}, which corresponds to a logical "1". The similar argument holds when x_1 and x_2 are low and y will be a logical "0". However, when one of the inputs is high and the other one low, for instance $x_1 = 1$ and $x_2 = 0$, N_1 and M_2 will be on. Assuming the transistors are properly sized y will be around $\frac{V_{DD}}{2}$ which does not correspond to any logical level. Therefore, this architecture picks the majority of its inputs, if exists. Otherwise, it produces a voltage level which is in the middle of logical voltage levels. This discussion may be extended for the case with four inverters easily.



(a) Two inverters with connected outputs



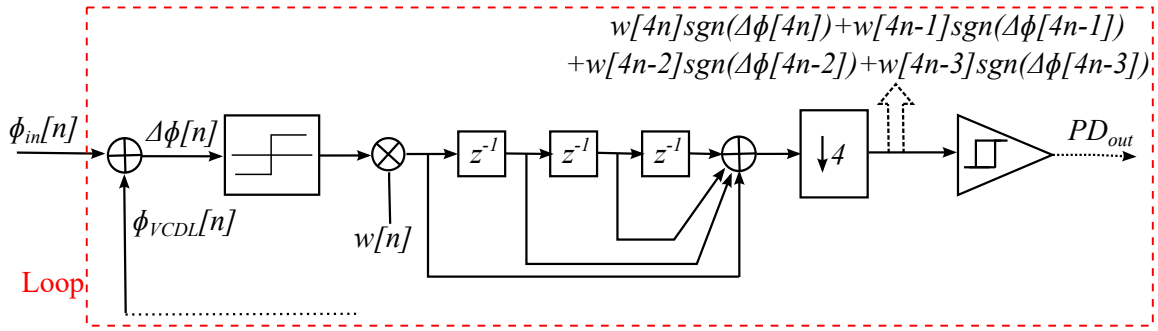
(b) Two inverters with connected outputs, transistor level

Figure 5.3: Averaging block

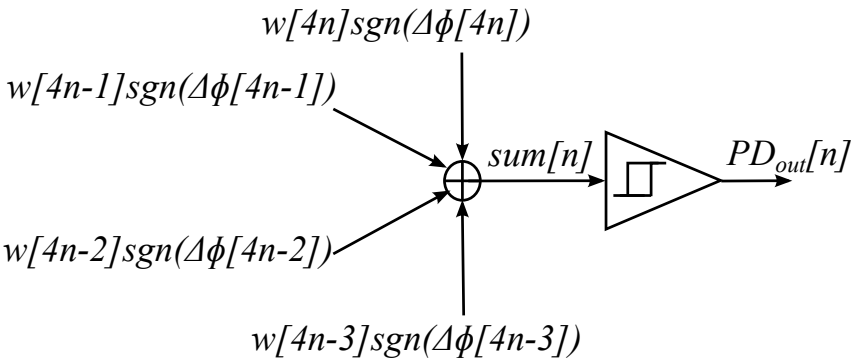
The averaging block's output is fed to a Schmitt Trigger. The advantage of the Schmitt Trigger over a simple slicer is its resilience to the noise at its input. Hence, according to this discussion, if there is a majority of $+1$ or -1 among the four outputs of averaging block, the PD output will be the same as the majority, otherwise the PD output will not change and

it retains its output value from the previous phase decision. This system block diagram is shown in Figure 5.4.

We can see this system performs two non-linear actions. We had studied the first one, which



(a) The phase detector complete block diagram



(b) The Schmitt Trigger only

Figure 5.4: Phase detector block diagram

is an Alexander phase detection in detail in Chapter 3, and developed a linear model for it. So as to be able to analyze this system we need to linearize the second non-linearity, which is different from the first one because of two reasons. Firstly, its input is not a continuous random variable. Secondly, if a majority does not exist the output is not determined by the input. Notwithstanding these differences, we can still use the methodology we deployed for BBPD linearization. We decompose the output of the Schmitt Trigger to a correlated part to input and an uncorrelated part. According to (4.5) we can define the gain as $K_{BBPD2} = \frac{E(sum \times PD_{out})}{E(sum^2)}$. The additive white quantization noise q_{PD2} also has the mean squared value of $\sigma^2_{q_{PD2}} = E(PD^2_{out}) - c^2E(sum^2)$.

In order to calculate these values we need to know the distribution of the input, *sum*. As is shown in Figure 5.4 (b):

$$\begin{aligned} sum[n] =& w[4n]sgn(\Delta\phi[4n]) + w[4n-1]sgn(\Delta\phi[4n-1]) \\ &+ w[4n-2]sgn(\Delta\phi[4n-2]) + w[4n-3]sgn(\Delta\phi[4n-3]) \end{aligned} \tag{5.1}$$

And as we proved in the appendix, in lock, $w[i]sgn(\Delta\phi[i])$ may be treated as an i.i.d. process. Therefore, we can convolve the PDF of each terms in (5.1) to find the PDF of $sum[n]$. Thus, we can easily find PDF of $w[i]sgn(\Delta\phi[i])$ by considering all cases:

$$P_{w[i]sgn(\Delta\phi[i])}(n) = \begin{cases} \frac{1}{4} & n = +1 \text{ or } -1 \\ \frac{1}{2} & n = 0 \\ 0 & \text{o.w.} \end{cases} \tag{5.2}$$

And from here we see that for the four independent contributions to the sum (see (5.1)):

$$P_{sum}(n) = P_{w[i]sgn(\Delta\phi[i])}(n) * P_{w[i]sgn(\Delta\phi[i])}(n) * P_{w[i]sgn(\Delta\phi[i])}(n) * P_{w[i]sgn(\Delta\phi[i])}(n) \tag{5.3}$$

This PDF is shown in Figure 5.5.

When the input is 0, which means that there is no majority, the output, PD_{out}, is determined

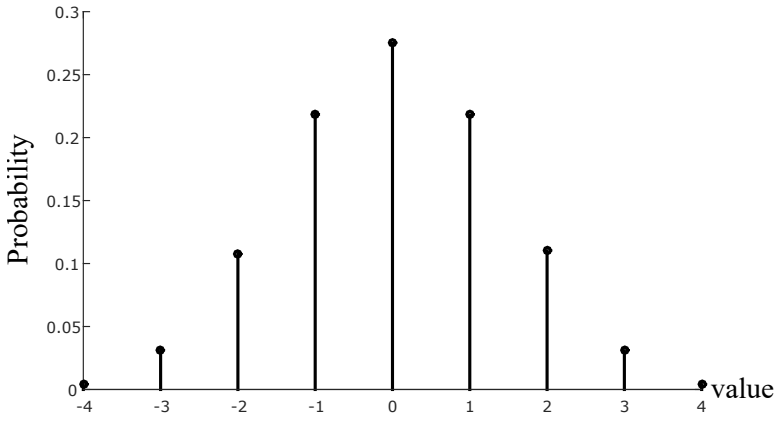


Figure 5.5: The PDF of the Schmitt Trigger input

by the previous output, which is independent from the current input. From here we can easily

find the gain:

$$K_{BBPD2} = \frac{E(PD_{out} \times sum)}{E(sum^2)} \tag{5.4}$$

First note that, due to Independence of $w[i]sgn(\Delta\phi[i])$ we can find the variance of sum using (5.1) and (5.2).

$$\begin{aligned} E(sum^2) &= 4 \times E(w[i]sgn(\Delta\phi[i])^2) \\ &= 4 \times \frac{1}{2} \\ &= 2 \end{aligned} \tag{5.5}$$

We can take two different approaches to calculate the gain, K_{BBPD2}. The first method, which calculates the exact value, finds $E(PD_{out} \times sum)$ by enumerating all possible cases with their corresponding probability from Figure 5.5.

$$E(PD_{out} \times sum) = \frac{35}{32} \tag{5.6}$$

Hence,

$$K_{BBPD2} = \frac{\frac{35}{32}}{2} = \frac{35}{64} \tag{5.7}$$

The other method is to use central limit theorem and approximate the PDF of sum with a Gaussian distribution[1] with the same variance (i.e. $\sigma = \sqrt{E(sum^2)} = \sqrt{2}$). We have found the BBPD gain for a Gaussian distribution in (4.8):

$$K_{BBPD2} \approx \sqrt{\frac{2}{\pi}}\frac{1}{\sigma} = \frac{1}{\sqrt{\pi}} \tag{5.8}$$

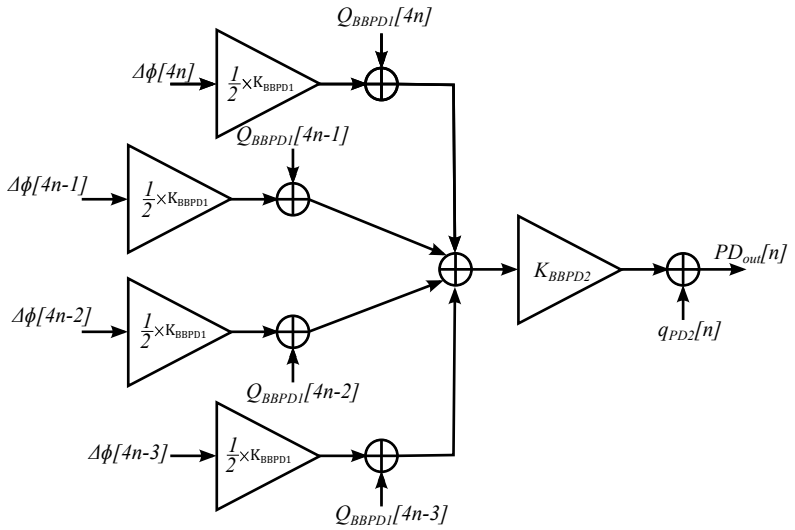
(5.7) and (5.8) are very close and match within 3%.

The additive quantization noise is:

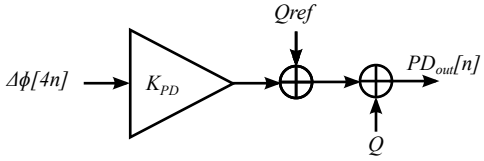
$$\sigma_{q_{PD2}} = PD_{out}^2 - \sigma_{sum}^2 K_{BBPD2}^2 = 1 - \left(\frac{35}{64}\right)^2 \times 2 \tag{5.9}$$

We can combine the linear model of the second non-linear operation in the phase detector with the linear model for the first non-linearity, to obtain a linear model for the complete

[1]Because sum is a linear super position of 4 independent random variables



(a) Four paths are linarized



(b) Viewing three paths as noise and one path as main

Figure 5.6: The PD linear model

phase detector, as shown in Figure 5.6.

The forward gain is the multiplication of the first and the second non-linear actions.

$$K_{PD} = \frac{1}{2} K_{BBPD1} K_{BBPD2} = \frac{1}{2}\sqrt{\frac{2}{\pi}}\frac{1}{\sigma_{\Delta\phi}} \times \frac{35}{64} \tag{5.10}$$

Although the addition of noises at the output in Figure 5.6 (b) , Q_{ref} and Q, act as an effective additive noise source, Q_{eff}. However, they originate differently.

Q_{ref} is the effect of the input noise which has travelled through the forward gain of the other three paths and appears at the output.

$$Q_{ref}[n] = \frac{1}{2} K_{BBPD1} \Delta\phi[4n-1] + \frac{1}{2} K_{BBPD1} \Delta\phi[4n-2] + \frac{1}{2} K_{BBPD1} \Delta\phi[4n-3] \tag{5.11}$$

Because of the i.i.d. assumption we can find the mean squared value of Q_{ref} by adding each

mean squared values:

$$\sigma^2_{Qref} = 3\left(\frac{1}{2}K_{BBPD1}\right)^2\sigma^2_{\Delta\phi} \tag{5.12}$$

Q encompasses the other noise sources: the bang-bang actions, and random sampling from all paths.

$$Q[n] = K_{PD2}(Q_{BBPD1}[4n]+Q_{BBPD1}[4n-1]+Q_{BBPD1}[4n-2]+Q_{BBPD1}[4n-3])+q_{PD2} \tag{5.13}$$

Again, owing to independence of each of the above terms:

$$\sigma^2_Q = 4K^2_{PD2}\sigma^2_{Q_{BBPD1}} + \sigma^2_{q_{PD2}} \tag{5.14}$$

5.3 Analysis parameters

With introduction of a linear model for the PD we are able to apply our theory to predict the output spectrum. However, first we must find the phase noise level at the loop input (Figure 5.1, left). We use the measured input jitter. It has been reported in the presentation slides[2] that the incoming data has $7.2ps$ peak to peak deterministic jitter and $2.6ps$ rms random jitter. We assume a Dual-Dirac distribution for the deterministic jitter, which corresponds to two impulses at $+3.6ps$ and $-3.6ps$ and from here the input jitter mean squared value σ_{in} is calculated. We also use the low ISI approximation.

$$\sigma_{in} = \sqrt{(2.6)^2 + (\frac{7.2}{2})^2} = 4.4ps \tag{5.15}$$

This time jitter, which is different from the phase noise by a factor of $\frac{2\pi}{T}$, is spread over the sampling Nyquist band of 20GHz. Therefore, it translates into a white phase noise with a white PSD of $-105dB/Hz$.

The following table was used for other loop parameters, as was provided by the authors, in a private communication.

[2]Available Online: http://ieeexplore.ieee.org/xpl/abstractMultimedia.jsp?arnumber=1696181

Analysis Parameters	
I_{CP}	$50\mu A$
R	200Ω
C	$79pF$
K_{VCO}	$870MHz/V$
K_w	$200rad^2.Hz$
BW	$25MHz$

Table 5.1: Loop simulation parameters

5.4 Analysis Results

In this section we compare the analysis results with reported measurements. Figure 5.7 compares the loop output spectrum from simulation and also the measurement. We can

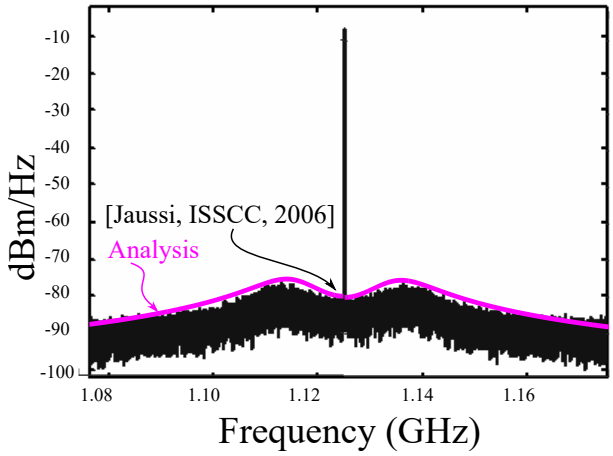


Figure 5.7: The loop output spectrum

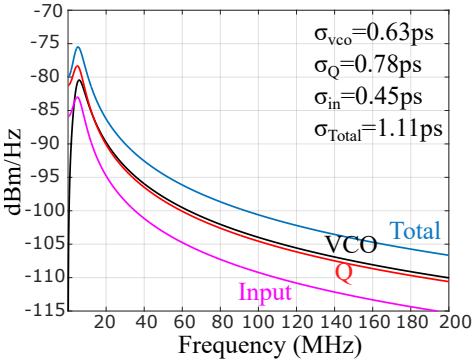
see the predicted spectrum matches the measurement very well. We can also calculate the rms output jitter by either integration of the predicted spectrum or the equivalent noise bandwidths.

The strength of our analysis is enabling us to see the break down of the contribution from each noise source (Figure 5.8). This can lead to design of an optimum loop. We investigated

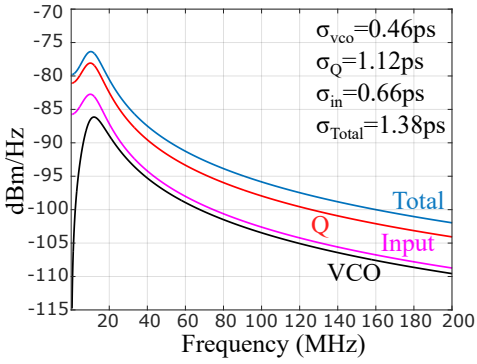
Method	RMS output jitter
Integration of PSD (Figure 5.7)	1.38ps
Calculation ((4.44))	1.29ps
Measurement	1.52ps

Table 5.2: Output jitter calculation

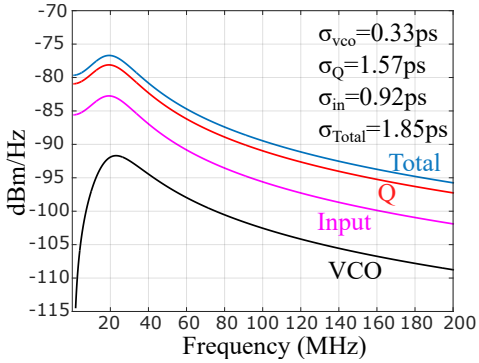
the loop performance for the measured loop filter as well as other loop filters. It turns out that the quantization noise is the most dominant noise source at the output. As Figure 5.8 shows, increasing the loop bandwidth increases the contribution from the input and quantization noise, and reduces the VCO contribution. In addition, The output jitter goes down with reducing the bandwidth and the best performance was achieved for the lowest bandwidth. However, further bandwidth reduction increases the VCO contribution and may not lead to a better output jitter.



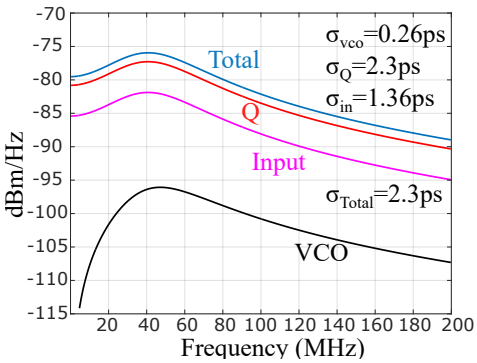
BW=10MHz



BW=25MHz, Measured spectrum



BW=50MHz



BW=100MHz

Figure 5.8: The loop output phase noise break down

CHAPTER 6

Conclusion

In this work we presented a frequency domain methodology for analysis and design of clock data recovery loops. The approach allows us to determine the noise contributions, compare them, and design the best loop for the application. It can reduce the reliance on time domain simulations and offer more intuition.

The random sampling of the phase detector was investigated and modeled by an effective gain and additive noise. We also studied the locking mechanism and mathematically showed how a tone at clock frequency is generated while the incoming data has no energy at that frequency.

Moreover, a linearized model for bang-bang phase detectors was presented, which is helpful for both CDR and PLL design. It is proved that this gain is determined by the incoming jitter jitter. By integrating this model with the linear random sampling model we are able to analyze any CDR loop, and predict the output phase noise accurately. We also illustrated the concept of equivalent noise bandwidth is applicable and we can simply predict the output jitter using it.

Appendix A

Correlation between $w[i]$ and $\Delta\phi[k]$

We know that due to ISI, phase error samples at a given time are influenced by prior bits. We noted that usually there is only one significant bit (for the case of Dual-Dirac jitter distribution) that determines this effect. It is reasonable to assume the most significant bit is the bit prior to each transition. Strictly speaking, the most significant transition is a more accurate name, because two possible cases for the deterministic jitter happen based on whether we had a transition prior to the current one or not; therefore, $w[i]$ and $\Delta\phi[i+k]$ are independent for $|k| > 1$.

Assume $\Delta\phi[k]$ can be decomposed into a deterministic component, which is d_1 if $w[k-1] = w[k] = 1$ and d_2 if $w[k-1] \neq w[k] = 1$, and a component due to the actual phase error $\Delta\phi_0[k]$.

$$\Delta\phi[k] = \Delta\phi_0[k] + d_1 w[k-1] + d_2(1 - w[k-1]) \tag{A.1}$$

Let us assume $f(\Delta\phi[k])$ has an average value m_f. Using this we can find the auto-correlation of $w[k]f(\Delta\phi[k])$.

$$R_{w\cdot f(\Delta\phi)}[n] = E\Big(w[k]f(\Delta\phi[k])w[k-n]f(\Delta\phi[k-n])\Big) \tag{A.2}$$

$$R_{w\cdot f(\Delta\phi)}[n] = \begin{cases} \frac{1}{2}m_f^2 & n = 0 \\ \frac{1}{4}m_f m_{f1} & |n| = 1 \\ \frac{1}{4}m_f^2 & |n| > 1 \end{cases} \tag{A.3}$$

In this equation m_{f1} is the average value of the f function if $\Delta\phi[k] = \Delta\phi_0[k] + d_1$ is the input (see Figure 3.12).

This auto-correlation can be expressed in another form:

$$R_{w\cdot f(\Delta\phi)}[n] = \frac{1}{4}m_f^2 + \frac{1}{4}m_f^2\delta[n] + \frac{1}{4}(m_f m_{f1} - m_f^2)(\delta[n-1] + \delta[n+1]) \tag{A.4}$$

To find the PSD we take the Fourier Transform,

$$S_{w\cdot\Delta\phi}(e^{j\omega}) = \frac{2\pi}{4}m_f^2 \sum_{k=-\infty}^{k=\infty} \delta(\omega + 2\pi k) + \underbrace{\frac{m_f^2}{4} + (m_f m_{f1} - m_f^2)(e^{j\omega} + e^{-j\omega})}_{\text{Noise}} \tag{A.5}$$

We can see the first term has an impulses at DC, and the second and third term are a colored noise. Thus, as we explained in Section 3.4 the tone at DC will drive the loop towards lock, which leads causes $m_f \to 0$. This result was derived earlier with an assumption of uncorrelation.

Now, consider the loop in lock (i.e. $m_f = 0$).

$$\begin{aligned} R_{w\cdot f(\Delta\phi)}[1] &= E\Big(w[k]f(\Delta\phi[k])w[k-1]f(\Delta\phi[k-1])\Big) \\ &= E(f(\Delta\phi[k-1]))E\Big(w[k]f(\Delta\phi[k])w[k-1]\Big) \\ &= 0 = E(w[k]f(\Delta\phi[k]))E(w[k-1]f(\Delta\phi[k-1])) \end{aligned} \tag{A.6}$$

Here we use the fact that $\Delta\phi[k]$ might have correlation with $w[i]$ only if $i < k$. Thus, we conclude that in lock, $w[i]$ and $\Delta\phi[k]$ are uncorrelated even for $|i-k| = 1$.

If the most significant transition was not the one prior to the current transition or we were dealing with a case with multiple impulses in ISI distribution, we could still use the similar approach and arrive at the same result.

References

[1] E. A. Lee and D. G. Messerchmitt, *Communication systems : an introduction to signals and noise in electrical communication*. Kluwer, 1988.

[2] A. B. Carlson, *Digital Communication*. McGraw-Hill, 1986.

[3] K. Mueller and M. Muller, "Timing Recovery in Digital Synchronous Data Receivers," *IEEE Transactions on Communications*, vol. 24, pp. 516–531, May 1976.

[4] R. Farjad-Rad, C. K. K. Yang, and M. A. Horowitz, "A 0.3-/spl mu/m CMOS 8-Gb/s 4-PAM serial link transceiver," *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 757–764, May 2000.

[5] J. Zerbe, C. Werner, V. Stojanovic, F. Chen, J. Wei, G. Tsang, D. Kim, W. Stonecypher, A. Ho, T. Thrush, R. Kollipara, G. J. Yeh, M. Horowitz, and K. Donnelly, "Equalization and clock recovery for a 2.5-10Gb/s 2-PAM/4-PAM backplane transceiver cell," in *Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International*, pp. 80–479 vol.1, Feb 2003.

[6] J. Sonntag, J. Stonick, J. Gorecki, B. Beale, B. Check, X.-M. Gong, J. Guiliano, K. Lee, B. Lefferts, D. Martin, U.-K. Moon, A. Sengir, S. Titus, G.-Y. Wei, D. Weinlader, and Y. Yang, "An adaptive PAM-4 5 Gb/s backplane transceiver in 0.25 mu;m CMOS," in *Custom Integrated Circuits Conference, 2002. Proceedings of the IEEE 2002*, pp. 363–366, 2002.

[7] Y. Takasaki, "Systematic Jitter Due to Imperfect Equalization," *IEEE Transactions on Communication Technology*, vol. 19, pp. 1275–1276, December 1971.

[8] J. Lee, K. S. Kundert, and B. Razavi, "Analysis and modeling of bang-bang clock and data recovery circuits," *IEEE Journal of Solid-State Circuits*, vol. 39, pp. 1571–1580, Sept 2004.

[9] J. L. Sonntag and J. Stonick, "A Digital Clock and Data Recovery Architecture for Multi-Gigabit/s Binary Links," *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 1867–1875, Aug 2006.

[10] M. Zanuso, D. Tasca, S. Levantino, A. Donadel, C. Samori, and A. L. Lacaita, "Noise Analysis and Minimization in Bang-Bang Digital PLLs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 56, pp. 835–839, Nov 2009.

[11] G. Marucci, S. Levantino, P. Maffezzoni, and C. Samori, "Analysis and Design of Low-Jitter Digital Bang-Bang Phase-Locked Loops," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, pp. 26–36, Jan 2014.

[12] Y. Choi, D.-K. Jeong, and W. Kim, "Jitter transfer analysis of tracked oversampling techniques for multigigabit clock and data recovery," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 50, pp. 775–783, Nov 2003.

[13] M. J. Park and J. Kim, "Pseudo-Linear Analysis of Bang-Bang Controlled Timing Circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, pp. 1381–1394, June 2013.

[14] T. Muoi, "Receiver Design for Digital Fiber Optic Transmission Systems Using Manchester (Biphase) Coding," *IEEE Transactions on Communications*, vol. 31, pp. 608–619, May 1983.

[15] A. H. Gnauck, G. Raybon, S. Chandrasekhar, J. Leuthold, C. Doerr, L. Stulz, A. Agarwal, S. Banerjee, D. Grosz, S. Hunsche, A. Kung, A. Marhelyuk, D. Maywar, M. Movassaghi, X. Liu, C. Xu, X. Wei, and D. M. Gill, "2.5 Tb/s (64 times;42.7 Gb/s) transmission over 40 times;100 km NZDSF using RZ-DPSK format and all-Raman-amplified spans," in *Optical Fiber Communication Conference and Exhibit, 2002. OFC 2002*, pp. FC2–1–FC2–3, Mar 2002.

[16] D. Breuer and K. Petermann, "Comparison of NRZ- and RZ-modulation format for 40-Gb/s TDM standard-fiber systems," *IEEE Photonics Technology Letters*, vol. 9, pp. 398–400, March 1997.

[17] G. Balamurugan, B. Casper, J. E. Jaussi, M. Mansuri, F. O'Mahony, and J. Kennedy, "Modeling and Analysis of High-Speed I/O Links," *IEEE Transactions on Advanced Packaging*, vol. 32, pp. 237–247, May 2009.

[18] S. Gondi and B. Razavi, "Equalization and Clock and Data Recovery Techniques for 10-Gb/s CMOS Serial-Link Receivers," *IEEE Journal of Solid-State Circuits*, vol. 42, pp. 1999–2011, Sept 2007.

[19] B. Casper and F. O'Mahony, "Clocking Analysis, Implementation and Measurement Techniques for High-Speed Data Links: A Tutorial," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, pp. 17–39, Jan 2009.

[20] "Understanding and Characterizing Timing Jitter." Tektronix, 2003 [Online]. Available: "http://www.tektronix.com/jitter".

[21] A. A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators," *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 1803–1816, Aug 2006.

[22] M. Shimanouchi, "An approach to consistent jitter modeling for various jitter aspects and measurement methods," in *Test Conference, 2001. Proceedings. International*, pp. 848–857, 2001.

[23] B. Analui, J. F. Buckwalter, and A. Hajimiri, "Data-dependent jitter in serial communications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 53, pp. 3388–3397, Nov 2005.

[24] J. Lee, P. C. Chiang, P. J. Peng, L. Y. Chen, and C. C. Weng, "Design of 56 Gb/s NRZ and PAM4 SerDes Transceivers in CMOS Technologies," *IEEE Journal of Solid-State Circuits*, vol. 50, pp. 2061–2073, Sept 2015.

[25] F. Gardner, "Self-Noise in Synchronizers," *IEEE Transactions on Communications*, vol. 28, pp. 1159–1163, Aug 1980.

[26] B. Razavi, *Design of Integrated Circuits for Optical Communications* . McGraw-Hill, 2003.

[27] W. R. Bennett, "Statistics of regenerative digital transmission," *The Bell System Technical Journal*, vol. 37, pp. 1501–1542, Nov 1958.

[28] B. R. Saltzberg, "Timing recovery for synchronous binary data transmission," *The Bell System Technical Journal*, vol. 46, pp. 593–622, March 1967.

[29] C. Hogge, "A self correcting clock recovery curcuit," *Journal of Lightwave Technology*, vol. 3, pp. 1312–1314, December 1985.

[30] J. D. H. Alexander, "Clock recovery from random binary signals," *Electronics Letters*, vol. 11, pp. 541–542, October 1975.

[31] D. L. Duttweiler, "The jitter performance of phase-locked loops extracting timing from baseband data waveforms," *The Bell System Technical Journal*, vol. 55, pp. 37–58, Jan 1976.

[32] E. Roza, "Analysis of Phase-Locked Timing Extraction Circuits for Pulse Code Transmission," *IEEE Transactions on Communications*, vol. 22, pp. 1236–1249, Sep 1974.

[33] "Fiber channel—Methodologies for jitter and signal quality specification-MJSQ," tech. rep., International Committee for Information Technology Standardization (INCITS), Mar. 10, 2003.

[34] Y. Cai, S. A. Werner, G. J. Zhang, M. J. Olsen, and R. D. Brink, "Jitter testing for multi-Gigabit backplane SerDes - techniques to decompose and combine various types of jitter," in *Test Conference, 2002. Proceedings. International*, pp. 700–709, 2002.

[35] "Jitter Analysis: The dual-Dirac Model, RJ/DJ, and Q-Scale." Tektronix, 2004 [Online]. Available: "http://cp.literature.agilent.com/litweb/pdf/5989-3206EN.pdf".

[36] M. H. Perrott, M. D. Trott, and C. G. Sodini, "A modeling approach for Sigma;- Delta; fractional-N frequency synthesizers allowing straightforward noise analysis," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 1028–1038, Aug 2002.

[37] N. D. Dalt, "Markov Chains-Based Derivation of the Phase Detector Gain in Bang-Bang PLLs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, pp. 1195–1199, Nov 2006.

[38] J. Jaussi, B. Casper, M. Mansuri, F. O'Mahony, K. Canagasaby, J. Kennedy, and R. Mooney, "A 20Gb/s Embedded Clock Transceiver in 90nm CMOS," in *2006 IEEE International Solid State Circuits Conference - Digest of Technical Papers*, pp. 1334–1343, Feb 2006.

[39] M. Y. He and J. Poulton, "A CMOS mixed-signal clock and data recovery circuit for OIF CEI-6G+ backplane transceiver," *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 597–606, March 2006.