Master Radiation and its Effects on MicroElectronics and Photonics Technologies (RADMEP)

Study of a Phase Locked Loop based Clock and Data Recovery Circuit for 2.5 Gbps data-rate

Master Thesis Report

Presented by

Stefano Marinaci

and defended at

University Jean Monnet

11/09/2023

Host Supervisor: Dr. Stefan Biereigel

Scientific Supervisors: Prof. Paul Leroux

Prof. Jeffrey Prinzie

Academic Supervisor: Dr. Arto Javanainen

Jury Committee: Prof. Sylvain Girard – University Jean Monnet

Dr. Arto Javanainen – University of Jyväskylä

Prof. Paul Leroux – KU Leuven

Prof. Frédéric Saigné – University of Montpellier

Title: Study of a Phase Locked Loop based Clock and Data Recovery Circuit for 2.5 Gbps data-rate

List of Abbreviations

ASIC	Application-Specific Integrated Circuit
BBPD	Bang-Bang Phase Detector
BB-PLL	Bang-Bang Phase Locked Loop
BER	Bit Error Rate
CDR	Clock and Data Recovery
CERN	Conseil Européen pour la Recherche Nucléaire
CMOS	Complementary Metal Oxide Semiconductor
COTS	Commercial Off-The-Shelf
CU	Capacitor Unit
DART28	Demonstrator ASIC for Radiation-Tolerant Transmitter in 28 nm
DCO	Digital Control Oscillator
EFM	Error Feedback Modulator
ENOB	Effective Number of Bit
FCC	Future Circular Collider
FCW	Frequency Control Word
FFT	Fast Fourier Transform
EFM	Error Feedback Modulator
GDS	Graph Data Science
GTM	Gigabit Transceiver Macromodel
HEP	High Energy Physics
HPC	High-Performance Computing
HL-LHC	High Luminosity Large Hadron Collider
LF	Loop Filter
LPF	Low Pass Filter
MASH	Multi-stAge noise SHaping
MFD	Most Frequent Decision
NCL	Noise Cancelling Logic
PLL	Phase Locked Loop
PD	Phase Detector
PI	Proportional – Integral
PRBS	Pseudo-Random Bit Sequence
PSD	Power Spectral Density
RTL	Register Transfer Level
SNR	Signal-to-Noise Ratio
ΣΔM	Sigma Delta Modulator
SEE	Single Event Effect
SSB	Single Side Band
TF	Transfer Function
TID	Total Ionising Dose
TMR	Triple Modular Redundancy
VCO	Voltage Control Oscillator

List of Figures

List of Tables

Abstract

Phase Locked Loop (PLL)-based Clock and Data Recovery (CDR) circuits play a big role in communication systems for High-Energy Physics (HEP) since are used to generate a high-quality clock signal that maintains synchronicity between the electronic systems. The CDR topology proposed in this thesis can be used to support the development of ASIC that finds application in high-speed link communication systems in the clock distribution system of the High Luminosity Large Hadron Collider (HL-LHC) detectors at CERN.

The thesis results from a study based on a theoretical analysis supported by simulations. This proposed work aims to describe the operation of a PLL-based CDR through a high-level behavioural analysis in Verilog. Advanced analyses are performed by using the Cadence Simulation Platform for modelling the system and Python scripting for requirement definition and post-processing of the data.

The dependency of the phase noise and jitter of the PLL is a topic that is relevant in systems designed for high-frequency synthesis, and the analysis of the jitter behaviour is required to minimize the noise contribution. During the design phase, each architectural choice comes with pros and cons, therefore high-level considerations are derived from this study. Different sources of noise are introduced and the effects on the CDR operation are studied to minimize the jitter contribution coming from the reference data stream and from the Digital Control Oscillator (DCO).

Furthermore, three CDR topologies that differ from the implementation logic of the downsampler connected at the output of the phase detector are compared. The comparison is done in terms of architectural complexity, bandwidth, jitter, and time to lock. As a result of the study, it is possible to conclude that a different implementation of downsampler logic generates a different phase detector gain in the feedback system, changing the dynamics of the system and its behaviour both in the time and frequency domains. However, it is possible to compensate for the variation of the phase detector gain by selecting different parameters for the digital filter, making in principle the behaviour of the three solutions equivalent.

Keywords: PLL-based CDR, High-speed Link communication systems, low-noise PLL.

Acknowledgements

First and foremost, I would like to express my deepest appreciation to my supervisor Dr Stefan Biereigel for his commitment and invaluable feedback. Without his assistance and dedicated involvement in every step throughout the process, this work would have never been accomplished.

I could not have undertaken this journey without Paulo Moreira and Szymon Kulis from CERN who generously provided knowledge and expertise for which I express my deepest gratitude.

I am also thankful to Prof. Paul Leroux and Prof. Jeffrey Prinzie from KU Leuven, and Dr. Arto Javanainen from JYU for the insightful comments, and conversations and for raising many precious points in our discussions that steered me in the right direction whenever needed.

This thesis stands also as a conclusion to this master. Therefore, I would also like to thank all the professors for sharing their passion and knowledge, teaching, and actively providing lots of useful lessons both inside and outside the class over these past years.

A special thank goes also to my classmates Theresa, Ruben, Felipe, and Cashmere with whom I shared a lot of experiences and concerns along this journey. I will carry with me a lot of positive memories.

A big thanks goes also to Leena, Amelie, and Inma, for their unwavering assistance at every stage, being always available and enthusiastic, and making this master a pleasant experience.

Thanks should also go to my colleagues Adam, Ludovic, and Mateusz for the inspiring talks and coffee breaks.

I would like to acknowledge the Erasmus+ Programme in the framework of RADMEP Master that provided the funding for the master scholarship, together with CERN as part of the Technical Student Programme.

Last but not least, none of this could have happened without my family. This dissertation stands as a testament to your unconditional love and encouragement.

Table of Contents

Introduction

The increasing request for higher-speed communication systems is driving the research to develop devices with better performance that can guarantee higher transmission speed, larger bandwidth, and higher data-rate.

To settle their requirements, communication systems such as high-speed wireline transceivers require very precise clock signals to drive their functional logic. In fact, they rely on accurate timing and precise clock signals.

One of the essential blocks that constitutes this technology is the frequency synthesizer, which as the name suggests is used to provide a clock signal at a targeted frequency. A Phase Locked Loop (PLL) is in general a preferred architecture for synthesizing a targeted clock frequency starting from a reference clock signal. PLLs can be used for various uses such as high-speed internet for data transmission, in satellites for accurate time synchronization, in logistics and High Energy Physics (HEP) electronics for timing distribution of the circuit or as jitter cleaning systems that reduce the noise of a noisy input signal, [2]. Therefore, depending on the applications, the requirements for each system can vary in terms of timing accuracy or noise sturdiness.

In particular, from an architectural point of view, All Digital Phase Locked Loop (AD-PLL) is a good alternative to synthesize a clock signal with limited jitter and a good quality clock signal. A digital PLL is a feedback system that requires a Digitally Controlled Oscillator (DCO) to provide a clock signal with a targeted oscillation frequency. An exhaustive description of PLL systems can be found in [1].

As a rule of thumb, more precise systems can be realized at the cost of higher power consumption to minimize noise levels and maintain speed performance. The quality of the generated clock signal depends therefore, not only on the quality of the design but also on the application context, where the system is used plays a big role to define the requirements. In fact, temperature variations or other external disturbances, such as radiation, can affect the performance of the system. In addition, even if PLLs are designed to provide good quality clock signals, the system itself can suffer from jitter and phase noise, either coming from the reference clock signals, that work as a reference timing source, or internally generated by the circuital components due to the nature of silicon-based technology which is made of. Therefore, when a PLL is designed, many aspects and constraints must be considered, such as the loop bandwidth, the range of the oscillation frequency that can be synthesized, and the locking time.

As mentioned, one limitation of these systems is represented by the phase noise and the jitter, due to the intrinsic nature of the electronics, which is bounded to temperature, corner variations and non-idealities in the architecture. In the first approximation, the noise sources of a PLL can be identified in two main contributions: the noise which comes from the reference and the noise generated by the DCO. Depending on the application, the requirements, or the environment where the equipment is used, it is possible to minimize the contribution of one source or the other. Even though a trade-off between the two should be chosen such that their contributions are balanced. In the next chapters, a more detailed analysis is presented to give a more accurate description of this concept.

A subset of PLL topologies is based on a Clock and Data Recovery (CDR) structure, which is a circuit that can generate a specific output clock signal starting from the data received from the reference signal under the assumption that the data received has a continuous 0/1 transition to trigger the clock generation, [3]. As reported, PLL-based CDRs play a big role also in communication systems used in High-Energy Physics (HEP) to provide a high-quality clock signal that establishes synchronicity between the

electronic systems. In this case, a big challenge is to design circuits that provide stable clock signals even in harsh environments where e.g., radiation is present. Hence, for systems used in particle accelerators, this becomes a relevant issue. For instance, the performance must be adapted to increase tolerance to radiation effects, like Total Ionising Dose (TID) effects and Singe Event Effects (SEE).

1. Application Context

High Energy Physics (HEP) experiments aim to measure events generated by collisions of particles at extremely high energies. For this purpose, enormous particle accelerators like the Large Hadron Collider (LHC) at CERN are built to study the fundamental interactions of elementary particles [4].

HEP experiments generate elementary particles that collide and interact. The energy generated by the collisions is then detected by a set of specially designed sensors. Multiple structured detectors are used to measure different properties of the particles generated in the collisions. The generated data are then stored for processing and analysis. In particular, experimental results can be compared with theoretical physics models to interpret and explain the origin of the Universe.

The electronics required for fulfilling these goals are extremely complex and challenging to build due to the highly demanding requirements. From an electronics point of view, high-speed integrated circuits for data transmission are required to provide high-speed communication to transmit the data collected by millions of detector channels and to store a huge amount of online data.

These electronic systems rely on high-speed communication systems (High-Speed links) that act as an intermediary between the sensors and the computing systems that store and analyse the data.

On the other hand, Commercial Off-The-Shelf (COTS) components usually do not satisfy the highly demanding requirements because most often these devices are used in harsh environments in which the high-energetic particles can cause different types of damage and errors. For this reason, is often required to explore and develop custom solutions to overcome such issues.

To improve the resolution of the measurement at the LHC, and to obtain more and more accurate results, a periodic series of upgrades is planned to reach higher energies and luminosities so that a higher occurrence of rare events can be measured. On the other hand, this makes the constraints more and more demanding and complex to attain. For instance, faster, more accurate and more performing electronics are required as well as radiation-tolerant devices for higher radiation levels. In particular, the High Luminosity Large Hadron Collider upgrade (HL-LHC) and the possibility to build even bigger particle accelerators such as the FCC (Future Circular Collider) [5, 24] are under study. The proposed system finds application in High-speed link communication transceivers for the HL-LHC.

2. Motivations

The increasing demands for high-performance devices, together with an increasingly harsh radiation environment and the trend of scaling CMOS technologies, make the design of new devices a real challenge.

The case study of this thesis targets to model a PLL architecture that generates a clock signal of 12.5 GHz starting from an input signal with a data rate of 2.5 Gbps. The study evaluates the effects of noise and minimizes the jitter contribution coming from the input data and the DCO.

The proposed CDR topology is mainly constituted by digital components so that the intrinsic advantages of digital circuits are exploited since they are less sensitive in terms of corner and process variations, as well as radiation, compared to their analog counterparts.

As an example, the proposed topology can be used to support the development of ASICs that find application in the clock distribution in systems used in the HL-LHC detectors, to reduce the clock jitter generated by a PLL-based CDR architecture. The high-speed link is a transceiver which interfaces the front-end detector with the High-Performance Computing (HPC) facilities used to perform the data analysis in the experiments, therefore it is expected to support high data rates and high-quality synchronicity. In particular, the Experimental Physics (EP) department at CERN has selected high-speed links as one of the key areas in the strategic R&D programme on technologies for future experiments, as explained in [6], [24].

The proposed thesis presents a high degree of innovation, not only for its competitive applications in HEP technology but also because the proposed system can be used in a harsh environment where radiation generated by the particle accelerators can pose a risk to the reliable operation of COTS devices. High synchronicity and clean clock signals are a must in the new generation of ASIC communication systems and are targeted for the successful fabrication of such systems. From a circuit point of view, this study allows exploring innovative solutions for the creation of high data-rate PLL-based CDR circuits. One of the main goals has been to analyse the requirements and a possible topology to implement a PLL-based CDR that starts from a data rate of 2.5 Gbps and can generate a clock signal of 12.5 GHz.

3. Design Approach

The proposed study is mainly realized using a behavioural model implemented in Verilog with Cadence Xcelium as a simulator and SimVision as a graphical Simulation Platform provided by Cadence to debug the code. This platform used as a circuit simulator, contains useful tools for the specifications and the requirement definition of the sub-blocks that constitute the circuit.

In addition, Python scripts have been developed for the requirement definition, for post-processing and data analysis, as well as to refine specifications of a defined architecture and compare the theoretical results with the data obtained by simulation. Furthermore, the effects of non-idealities, such as noise sources or delays are analysed considering their contributions separately or their overall effects on the modelled components as ideal and noise-free blocks. This approach involves the definition of an initial abstract model in Verilog to evaluate the technical feasibility, the specifications, and the most critical aspects of each sub-block before performing a system-level simulation with a limited computational burden. Several critical aspects and non-idealities such as the jitter, and phase noise are considered. In particular, each sub-block, which is part of a more complex system, is individually simulated and tested to substantiate the design choices made during the modelling phase. The main challenges in this step come from the definition of what are the constraints of each subblock in terms of low noise and high accuracy in timing.

The work presented in this thesis strongly relies on the study of parameters and figures based on mathematical calculations and functions, such as the Fast Fourier Transform (FFT) and the Power Spectral Density (PSD). In particular, parameters such as phase noise, sidebands, spurious components, and jitter, are extracted to evaluate the performance of the system. Furthermore, when analysing the quality of a clock signal, it is important to study not only its behaviour in the time domain but also the frequency domain to provide enlightened information such as the transmission bandwidth, the source of noise and how this affects the system performance.

4. Thesis Structure

The thesis is structured as follows:

After a concise Introduction, Chapter 1 provides basic information about the operation of the CDR and presents the last trends in the state-of-the-art of PLLs.

Chapter 2 presents a preliminary analysis of the linear time-invariant system by using the phase transfer function and studying how the noise from different sources propagates along the system.

Chapter 3, shows a block diagram representation of the proposed PLL-based CDR circuit, analysing each block and how these are correlated. In particular, the proposed architecture is constituted by a chain of digital blocks such as a Bang-Bang Phase Detector (BBPD), a Downsampler, a 1st order proportional, integral digital filter, 2nd order Sigma Delta Modulator (MASH 1-1), LC Digital Control Oscillator (LC-DCO), and a frequency divider. In addition, the performance of the Digital PLL architecture is evaluated.

Chapter 4 compares three similar topologies where the downsampler is the sub-block that operates differently. In particular, the operation logic of this block is implemented with a different algorithm, such that the effects of the overall CDR architecture can be studied. The comparison is done in terms of Phase Detector (PD) gain, jitter and dynamic of the CDR taking into account different filter parameters.

Lastly, the general conclusions of the overall study are outlined.

Chapter 1 - Background

In this chapter, some basic knowledge and considerations about the operation of a CDR are summarized as well as a literature review is given to highlight some of the most relevant works studied to better understand the topic and to provide essential insights to understand the resulting consideration derived hereafter.

PLLs are used in many applications, for example, to synthesize a targeted clock frequency, for timing distribution in the circuit, as a jitter cleaner to reduce the noise of a noisy input signal and depending on its application, structural variations in the topology are implemented.

However, the working principle of a PLL, at the system level can be modelled considering the main building blocks. The operation can be described through parameters such as the operating frequency range, which is the range of frequencies that PLL can generate, the lock time, which is the time a PLL needs to lock on the reference frequency the jitter, that is the undesired deviation from an ideal variation of a periodic signal.

This thesis aims to describe the block diagram and the operation of an All-Digital PLL. These architectures provide many advantages compared to their analog counterparts such as reduced power consumption and higher robustness and lower sensitivities to corner and process variations as well as the fact that they can be fully synthesizable, so they can be easily customized and implemented from Register Transfer Level (RTL) to Graph Data Science (GDS). Also, they require a much smaller occupied area, to ensure a larger level of noise resilience (sub-ps jitter) last but not least the implementation of radiation hardening techniques such as Triple Modular Redundancy (TMR) of the logic is relatively simple to be added, [3].

In high-speed communication systems, the phase noise, which is the unwanted fluctuation in the phase of a signal w.r.t an ideal clock signal, is targeted to be as low as possible and is expressed in units of dBc/Hz. This effect in the time domain translates into small fluctuations in the phase of the signal around the ideal clock signal. Phase noise is also strictly related to jitter since a random variation in the phase of a signal translates into a fluctuation of the edges of a clock signal concerning the ideal edge and is generally expressed in ps.

When observed in the frequency domain, through the Fast Fourier Transform (FFT), the phase noise is seen as a spread of the tones of the harmonics and in a higher noise floor which causes a deterioration in the system performance and a higher Bit Error Rate (BER) with consequences such as data corruption.

In a PLL there are many sources of jitter and in some cases, depending on the source that generates it, their contribution can be more prominent at lower frequencies or higher. Therefore, the application as well as the environment can corrupt the performance of the system. This chapter describes the basic operation of a PLL and the state-of-the-art of some of the most common techniques used to overcome the limitations of currently existing PLLs.

1.1 Basic Operation of PLL

The basic operation of a PLL emerges from the combined operation of a few main building blocks: a Phase Detector (PD), a Low Pass Filter (LPF), a Voltage-Controlled Oscillator (VCO) and a frequency divider. A simplified block diagram is given in Figure 1.1, in which just the main blocks are considered. In particular, the PD has the goal to match the phase of a reference signal, modulated with a specific frequency with the signal generated by the VCO after is divided by the divider. However, due to the noise and other non-idealities of the system, the two signals are never perfectly matched.

However, since it is based on a feedback structure, the PLL corrects for any differences in frequency and phase to minimize them. Once the phase difference is minimized, the PLL is locked such that the oscillation frequency produced by the VCO is proportional to one of the reference clocks. According to the block diagram in Figure 1.1, the first block in the chain is the PD, which measures the difference in phase of the two input signals. The error measured is then filtered by the Low Pass Filter (LPF). The signal generated by the LPF is then used to control the frequency generated by the VCO. The oscillation frequency of the VCO is slightly increased or decreased around a targeted oscillation frequency. The frequency divider ensures that the reference and feedback clock signals arriving at the phase detector have the same frequency, which allows their phase to be compared.

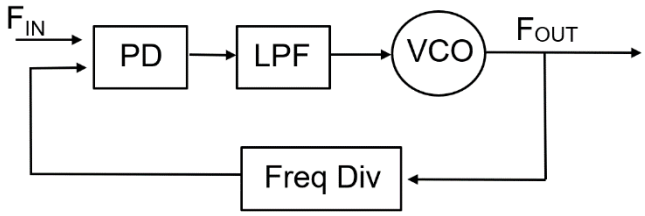

Figure 1.1: Simplified Block diagram of a PLL.

1.2 State-of-the-Art

Many interesting studies have been proposed in the literature, in which several topologies and useful techniques for PLLs are examined. Among these, some are particularly interesting to understand the study presented in this thesis since they provide enlightening discussion regarding some aspects of the operation of a PLL.

Ref. [7] describes the pros and cons, in terms of phase noise, of using a dithering technique with a 2nd order Sigma Delta Modulator (ΣΔM), [7]. The ΣΔM is useful to lower the quantization noise introduced by the oscillator in a PLL. In particular, the frequency resolution of the DCO is analysed. The 2nd order Multi-stAge noise SHaping (MASH) dithering shifts the quantization noise introduced by the capacitor array to high-frequency and the advantages are investigated.

An exhaustive description of a PLL-based CDR for Serial Data transmission based on a Bang-Bang Phase Detector (BBPD) and a 2nd order loop in which a 1st order filter is used in Ref. [8]. As a result, this paper gives a deep explanation of what the operation of the nonlinear Bang Bang-Phase Locked Loop (BB-PLL) looks like. In particular, the dynamics of the feedback system provide information about the jitter, and the spectrum of the generated output and provides insight into the behaviour of the system as a function of various parameters.

In the PhD thesis by Bueren, G. [9], the design of a Clock and Data Recovery Circuit are presented, giving more emphasis on critical and speed-limiting circuits. The main focus of this work is the study of how the data stream is used to generate a clock signal and how the jitter at the output of the transceiver, and the jitter coming from the sampling clock, affect the performance of the system.

A comprehensive study has been given in Ref. [10], which focuses on the design and the study of radiation-tolerant clock generation circuits that are used in HEP experiments. In particular, the challenges and the effects of electronics to increase the reliability and the behaviour of circuits used for HEP are studied. Several techniques and tool

configurations are developed and analysed to understand the design challenges in submicron CMOS nodes. Furthermore, this work focuses on both low-speed PLL/CDR applications and the high bit-rate all-digital CDRs. The second is indeed complementary to the study presented in this thesis.

For a better understanding of the problems that affect a PLL, some application notes such as Refs. [11], [12], [13,] are interesting. In particular, some basic definitions such as phase noise, and jitter are given, as well as the derivation of some fundamental calculations.

In addition, many useful considerations from Ref. [14] are fundamental to better understanding the operation of digital BB-PLL used for high-speed serial data communications systems. The main challenges experienced during the design phase are the chip integration, as well as low jitter, high bandwidth, and high robustness against noise in low-supply applications, are summarized.

The design process of a Clock and Data Recovery (CDR) circuit used at Gbps rates is described in Ref. [15]. In particular, the features of the system and the trend of developing systems with improved performance are highlighted. For this reason, more effective solutions need to be used to maintain low power, high robustness to supply variation and to process technology. In addition, some considerations on the CDR implementation are obtained while comparing a Phase Detector (PD) and Charge Pump (CP) at the data-rate speed and another that operates at a lower rate.

Chapter 2 - Transfer Function Analysis

2.1 Introduction

PLLs can be used to generate good-quality clock signals, however, jitter and phase noise, represent undesired effects to take into account when designing a PLL and defining its requirements such as the tuning range, the locking time, and the loop bandwidth. The proposed study is referred to as an All-Digital PLL (AD-PLL), which is constituted by digital components which include also the oscillator. For this reason, the oscillator will be referred to as a Digital Control Oscillator (DCO), instead of the more conventional Voltage Control Oscillator (VCO).

Depending on the application, and the environment where a PLL is used, might be necessary to suppress noise contributions at lower or higher frequencies. The study of the transfer function represents a useful approach to understanding how the noise propagates through the PLL. In this chapter, a detailed analysis aims to give a more accurate description of this concept.

The key aspect is to understand the behaviour of a PLL, indeed by changing the PLL bandwidth, it is possible to change the amount of jitter in the feedback system. In practice, this is done by changing the Low-Pass Filter (LPF) bandwidth. Generally speaking, the bandwidth should be chosen such that both the DCO and the reference jitter are balanced and optimized according to the application. Hence, if the system has a larger noise contribution coming from the DCO, a larger bandwidth is preferred. The jitter from the reference clock can vary depending on the application and the setup used, while the jitter contribution from the DCO is mainly defined by the noise contribution coming from the circuit components together with the quality of the design and the layout of the circuit.

2.2 Analysis of the Noise Behaviour in the Feedback System

A preliminary analysis in terms of transfer function is derived, to describe a linear time-invariant system. Even though the PLL is constituted by nonlinear circuits, such as the PD, its operation is approximated by a linear model, provided that the phase mismatch is small and the PLL is locked. Therefore, Laplace and Fourier Transforms can be used to perform the linear analysis. More specifically, for the PLL, the input/output is expressed in terms of the phase of the signals rather than as voltages and/or currents as in conventional transfer functions, therefore this is considered a phase transfer function analysis [25]. This analysis will be combined with the main noise contributions of the PLL, which are the DCO noise and the reference noise, to study their behaviour and understand how modelling the dynamics of the PLL is beneficial to minimize the noise sources. Each main functional block is modelled as a black box from which the transfer function is derived.

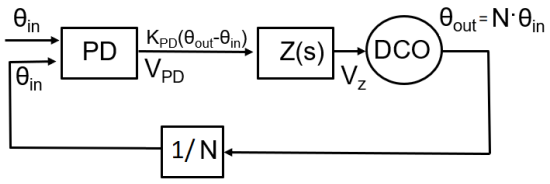

Figure 2.1: Simplified representation of the block diagram for the phase transfer function analysis.

The following analysis is referred to the Figure 2.1 which shows a closed-loop PLL for which the phase transfer functions of the main component blocks are discussed in this

section. In terms of phase, $\theta_{out}(t)$ represents the output phase of the signal generated by the oscillator, while $\theta_{in}(t)$ is the phase of the input signal of the Phase Detector (PD). Both signals represent the input of the PD that provides an output voltage proportional to their difference in phase:

$$v_{PD} = K_{PD}(\theta_{out} - \theta_{in}) \quad (2.1)$$

where K_{PD} is the gain of the PD. The measured phase difference is considered a phase error and generates a proportional voltage V_{PD} at the output of the PD. The signal generated by the PD is then filtered by a loop filter Z(s), whose output signal V_z (s) in the Laplace domain, as a product of the filter, Z(s), and the output of the PD, V_{PD} (s):

$$V_z(s) = Z(s) \cdot V_{PD}(s) \quad (2.2)$$

The next building block is the VCO, which behaves as an integrator, which takes as input a frequency control word proportional to the phase error measured at the input of the PD and generates an output frequency accordingly. Is worth reminding that the frequency is the rate of change of a phase $\left(\frac{d\theta}{dt} = \omega\right)$ and the integral of the phase, θ, is the frequency, ω. The DCO transfer function has a high-pass filter shape that can be modelled as follows:

$$\theta_{out} = \frac{K_{DCO} \cdot V_Z(s)}{s} \quad (2.3)$$

where K_{DCO} is the oscillator gain, N is the division ratio provided by the frequency divider and $s = \sigma + j\omega$ is the complex frequency in the Laplace domain. Thus, the output phase of the DCO is proportional to the integral of the control voltage. Once the basic blocks of the system are defined, it is possible to derive the overall transfer function of the system. A detailed model analysis is also available in the literature, e.g. [1].

The open loop transfer function (TF_{OL}) is obtained by combining the PD, the filter and the oscillator transfer function (TF_{DCO}) such that it represents the cascaded behaviour of each component:

$$TF_{OL}(s) = \frac{K_{PD} \cdot K_{DCO} \cdot Z(s)}{s} \quad (2.4)$$

The phase errors are modelled by the DCO transfer function which is defined as:

$$TF_{DCO}(s) = \frac{1}{1 + TF_{OL}(s) \cdot \frac{1}{N}} \quad (2.5)$$

Combining (2.4) and (2.5), the DCO transfer function results:

$$TF_{DCO}(s) = \frac{s}{s + \frac{K_{PD} \cdot K_{DCO}}{N} \cdot Z(s)} \quad (2.6)$$

The PLL is modelled as a closed loop system, therefore, the transfer function (TF_{CL}) is obtained by substituting (2.4) with the closed loop transfer function:

$$TF_{CL}(s) = \frac{TF_{OL}(s)}{1 + TF_{OL}(s) \cdot \frac{1}{N}} = \frac{K \cdot Z(s)}{s + K \cdot Z(s)} \quad (2.7)$$

where $K = \frac{K_{PD} \cdot K_{DCO}}{N}$.

In principle, depending on the requirements of the PLL, it is possible to select a filter with higher selectivity to further reduce high-frequency contribution. In particular, Z(s) can be a 1st order filter as in this case, but if needed a higher-order filter can be used as well.

2.3 Reference Noise Measurement

To further proceed with the study using realistic data, some reference noise samples are measured. In this case, a clock signal is used to measure the phase noise, however, according to its application, the PLL-based CDR is expected to operate under the assumption that the data received has a continuous 0/1 transition to trigger the clock generation. The measurement of reference noise is taken using FPGA Xilinx UltraScale with GTM (Gigabit Transceiver Macromodel) transmitter at multiple frequencies using:

- Copper direct attached cable (coaxial cable).
- Fiber cable (QSFP 25 Gbps NRZ optical transmitter module).

The reported measurements in Figure 2.2, are obtained using clock frequencies of 1.28 GHz and 3.2 GHz. In both cases, the phase noise measured for different carrier frequencies is normalized to that of 2.5 GHz in Figure 2.2.

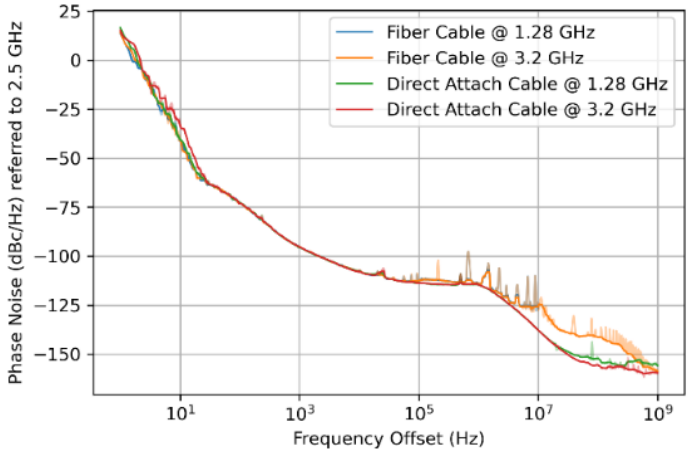

Figure 2.2 Reference noise sample measurements taken using both copper cable and fibre cable with different carrier frequencies.

	Jitter (fs)
Fiber Cable @ 1.28 GHz	407.34
Fiber Cable @ 3.2 GHz	425.86
Direct Attach Cable @ 1.28 GHz	314.48
Direct Attach Cable @ 3.2 GHz	313.22

Table 2.1: Jitter calculation for the reference noise measurements in Figure 2.2.

By comparing the two curves, the copper cable provides less noise than the fibre cable at a higher frequency, therefore for short-distance high-frequency transmission, the copper cable is preferable, but the reference noise obtained with fiber cable will be used during the simulation to provide a setup for worst case scenario analysis. The jitter contribution in Table 2.1 from the reference noise is calculated from the curves in Figure 2.2.

2.4 Oscillator Phase Noise - Leeson's Equation

In this section, the phase noise of the DCO is modelled. The phase noise in the oscillator is produced by fluctuations in the phase of the signal caused by e.g., thermal noise, shot noise and flicker noise on circuit components. The phase noise generated by a PLL can be modelled by Leeson's equation which represents the spectrum of the oscillator phase noise, expressed as the PSD of the single-sideband (SSB) phase noise in dBc/Hz. A preliminary step for this study aims to model the DCO parameter starting from some

realistic simulation data. This approach will allow to estimate some parameters to fit the DCO phase noise to Leeson's equation. In particular, all three regions are separately considered, starting from the noise floor as the base one and adding the others.

A simplified version of Leeson's Equation, ($\mathcal{L}_{app}(f)$), gives a preliminary estimation of the noise behaviour in the VCO, since considers the two main noise contributions: the noise floor and $1/f^2$. A more detailed explanation can be found in ([1], [28]). Leeson's Equation, ($\mathcal{L}_{app}(f)$) is modelled as:

$$\mathcal{L}_{app}(f) = 10 \cdot log_{10}\left(\frac{1}{2} \cdot \frac{F \cdot \mathrm{k} \cdot \mathrm{T}}{P}\left(\frac{f_{VCO}}{2 \cdot Q_L \cdot \mathrm{f}}\right)^2\right) \quad (2.7)$$

where f_{VCO} = 12.5 GHZ is the oscillation frequency of the oscillator, k ≈ $1.38 \cdot 10^{-23}$ J/K is the Boltzmann's constant, T = 300K is the operation temperature, F is the Noise Figure of the active device considered, P = 10mW is the RF power at the input of the active device, $Q_L = (2 \cdot \pi \cdot f_{VCO} \cdot L)/R$ is the Loaded Quality Factor obtained with the inductor L and R that is the resistance of the inductor.

An extended version of Leeson's equation models the phase noise in three regions: $1/f^3$, $1/f^2$, and noise floor. In particular, the noise floor gives a flat contribution and is dominant at high frequencies. The white noise at the input of the oscillator is transformed at the output of the oscillator as $1/f^2$, which is the result of the integration of the noise power. This contribution has a slope of 20 dB/dec. In addition, the Flicker nose, when integrated by the oscillator gives origin to the $1/f^3$ noise which presents a slope of 30 dB/dec. The extended Leeson's equation ($\mathcal{L}_{ext}(f)$), as reported in [28], is modelled as:

$$\mathcal{L}_{ext}(f) = 10 \cdot log_{10}\left(N_3 \cdot \left(\frac{f_{def}}{\mathrm{f}}\right)^3 + N_2 \cdot \left(\frac{f_{def}}{\mathrm{f}}\right)^2 + N_0\right) \quad (2.8)$$

where the noise contribution is expressed as a linear combination of three components: $1/f^3$, $1/f^2$, and noise floor. The three coefficients are expressed as:

$$Noise_{\frac{1}{f^3}} = N_3 = \frac{F \cdot \mathrm{k} \cdot \mathrm{T}}{P}\left(\frac{f_{1/f^3} \cdot {f_{VCO}}^2}{8 \cdot {Q_L}^2 \cdot {f_{def}}^3}\right)^2 \quad (2.9)$$

$$Noise_{\frac{1}{f^2}} = N_2 = \frac{F \cdot \mathrm{k} \cdot \mathrm{T}}{P}\left(\frac{{f_{VCO}}^2}{8 \cdot {Q_L}^2 \cdot {f_{def}}^2}\right)^2 + \frac{2 \cdot \mathrm{k} \cdot \mathrm{T} \cdot R}{{f_{def}}^2} \quad (2.10)$$

$$\text{Noise floor} = N_0 = \frac{F \cdot \mathrm{k} \cdot \mathrm{T}}{P} \quad (2.11)$$

where f_{1/f^3} in (2.9) is the corner frequency between $1/f^3$ and $1/f^2$, f_{def} is the normalized offset for phase noise, R is the resistance of the DCO, while the other coefficients are mentioned above.

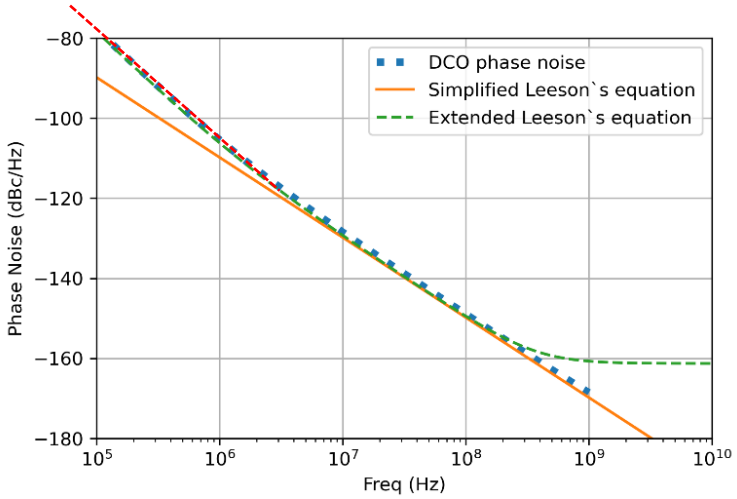

Figure 2.3: Leeson's equation matching the DCO simulation model.

As shown in Fig.2.3 three regions are separated respectively at the frequency of 1.3 MHz as the corner frequency between $1/f^3$ and $1/f^2$ and 300 MHz as the corner frequency between $1/f^2$ and the noise floor. Figure 2.3 shows an example of a DCO phase noise at which the simplified and extended Leeson's equations fit the noise behaviour of the oscillator. In particular, the DCO phase noise is taken from the simulation of the current DART28 (Demonstrator ASIC for Radiation-Tolerant Transmitter in 28 nm) PLL circuit [10].

For a more accurate estimation of the phase noise of an oscillator, an extended version of Leeson's equation is preferred. The contribution of $1/f^3$ noise is highlighted by the red dashed line. Leeson's equation will be used for successive analysis to understand how the noise propagates through the PLL and how it its shape changes depending on the operation condition of the PLL.

2.5 Effects of Dithering Introduced by a ΣΔ Modulator on the Phase Noise

Due to the discretized number of capacitors used to control the oscillation frequency of the DCO, a limited number of possible oscillation frequencies can be generated. This translates into quantization noise. The quantization noise can be a relevant contribution when low-phase noise is targeted.

In principle, a large number of small capacitors should be used to control the oscillation frequency of the DCO, which reduces quantization noise. The smaller the capacitor the finer the attainable tuning. However, depending on the tuning range and the oscillation frequency required, the number of capacitors is bounded by the minimum capacitor size attainable. Therefore, a smaller unit capacitor requires a larger number of capacitors needed to tune the same range, and parasitics become dominant in the design. A possible approach to reduce the quantization noise coming from the DCO is to use a Sigma Delta Modulator (ΣΔM), as reported in [7]. A ΣΔM can indeed reduce the phase noise since it moves the quantization noise to a higher frequency with a slope of +20dB/dec per order. Indeed, a larger set of capacitors can be used since fine-tuning is taken care of by the ΣΔM which takes as input the fractional control word and in return generates an average of its input signal. On the other hand, the ΣΔM can also introduce noise to the system. Therefore, it is important to choose appropriate ΣΔM parameters, such as operation frequency, resolution, and order. In this section, a study, which is based on [7], is reported to choose the best trade-off among these parameters.

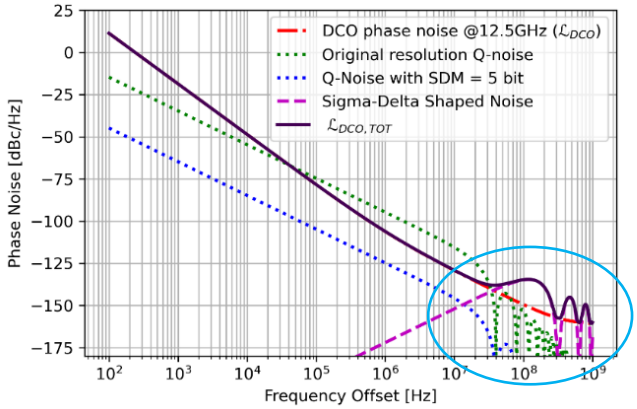

Figure 2.4: Representation of the total phase noise when a 2nd order ΣΔM with $f_{\Sigma\Delta M}$ = 320 MHz is added $L_{DCO,\,TOT}$ [dBc/Hz].

Figure 2.4 shows the different contributors to the oscillator phase noise spectrum for a DCO driven by a ΣΔM. In particular, the quantization noise of the DCO, (dotted green line) is higher than the DCO phase noise (dashed red line) for offsets between 100 kHz and 30 MHz. Therefore, the use of a ΣΔM can be useful to reduce the quantization noise. As a result, the quantization noise, $\mathcal{L}_{QNOISE}$, (dotted blue line) is reduced. The shaped noise, $\mathcal{L}_{\Sigma\Delta M,}$ (dashed magenta line) at higher frequencies must be also added to the overall contribution. The total noise ($\mathcal{L}_{DCO,\,TOT}$) contribution is calculated in (2.12) and in Figure 2.4 is shown (dark purple continuous line).

$$\mathcal{L}_{DCO,\,TOT}\,[\text{dBc}] = \mathcal{L}_{DCO} + \mathcal{L}_{QNOISE} + \mathcal{L}_{\Sigma\Delta M} \quad (2.12).$$

The integration range considered for DCO jitter calculation is [100 kHz – 1 GHz]. Some considerations should be evaluated to understand which is the optimal ΣΔM configuration in terms of jitter, operation frequency and power consumption. As a rule of thumb, a higher operation frequency ($f_{\Sigma\Delta M}$) corresponds to a shift of the noise to a higher frequency, which is preferable. On the other hand, a circuit operating at high speed also brings a higher power consumption which is undesirable. In particular, for this example to minimize the quantization noise of the DCO a ΣΔM with the following features can be used: a 2nd order ΣΔM with a resolution of 5 bit and operation frequency $f_{\Sigma\Delta M}$ = 320 MHz. To compare the two solutions, shown in Figures 2.4 and 2.5, the jitter is calculated for both configurations such that:

- $f_{\Sigma\Delta M}$ = 320 MHz, Total jitter ($\mathcal{L}_{DCO,\,TOT}$) = 505.63 fs.
- $f_{\Sigma\Delta M}$ = 1.25 GHz, Total jitter ($\mathcal{L}_{DCO,\,TOT}$) = 504.12 fs.

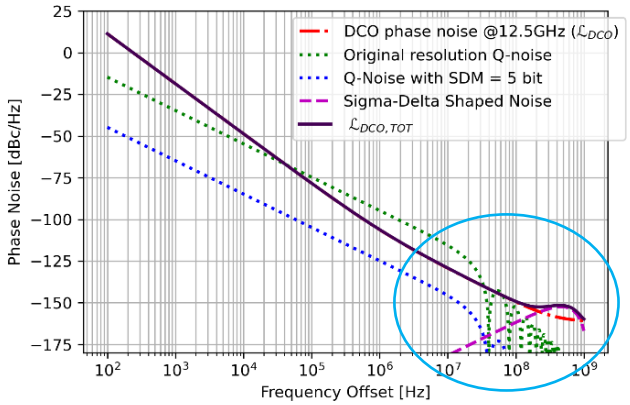

Figure 2.5: Phase noise estimation for a 2nd order ΣΔM with $f_{\Sigma\Delta M}$ = 1.25 GHz.

As a result, by increasing the operation frequency, the noise shape is moved to a higher frequency which in principle is a good choice, as shown in Figure 2.5, however, high operation frequency causes an increase in power consumption for only a small improvement in terms of jitter.

Significant improvements can be obtained with a different ΣΔM topology. In fact, a higher order ΣΔM offers a higher selectivity in the noise shaping, which is expected to correspond to +20dB/dec for a 1st order, +40dB/dec for a 2nd order, +60dB/dec for a third order modulator. Therefore, a higher order ΣΔM should be more advisable, but on the other hand, it comes at the cost of a more complex topology and computational operation which is unfavourable. On the other hand, a lower order generates lower jitter at a lower frequency, but increases the spurious tones due to a limited number of sequences generated.

Another consideration derived from this analysis is related to the advantages and drawbacks of using a 2nd-order or a 3rd-order ΣΔM with the same operation frequency ($f_{\Sigma\Delta M}$ = 320 MHz), shown in Figures 2.4 and 2.6. Also, in this case, the comparison is done in terms of jitter for different ΣΔM orders. As a result, from the calculation is it possible to show that:

- 3rd order ΣΔM Jitter, Total jitter ($\mathcal{L}_{DCO,TOT}$) = 508.79 fs.

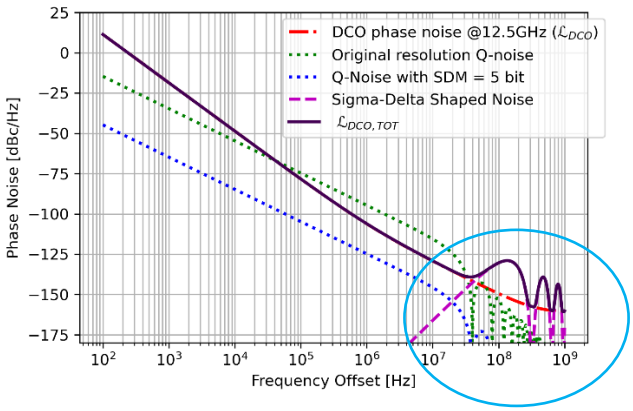

Figure 2.6: Phase noise estimation for a 3rd order ΣΔM with $f_{\Sigma\Delta M}$ = 320MHz.

A 3rd order ΣΔM provides a more selective noise shaping at High Frequency (HF) at the cost of higher jitter.

Jitter : $\mathcal{L}_{DCO,TOT}$ [fs]						
f_{SDM}	40MHz	80MHz	160MHz	320MHz	640MHz	1280MHz
ΣΔ order = 1	555	517	507	504	504	504
ΣΔ order = 2	602	530	510	**505**	504	504
ΣΔ order = 3	761	578	523	508	505	504

Table 1.2: Jitter calculation for ΣΔM order and operation frequency ($f_{\Sigma\Delta M}$). The values are given in femtoseconds.

As a result:

- 1st order ΣΔM generates lower jitter but on the other hand provides additional spurious tones due to a limited number of output sequences.
- 2nd order ΣΔM with 320 MHz $f_{\Sigma\Delta M}$ seems the optimal configuration.
- 3rd order ΣΔM generates higher jitter at low $f_{\Sigma\Delta M}$ at the cost of a more complex system.

2.6 First-Order Loop Filter Modelling

In this section, the features of a filter are defined. In fact, depending on the requirements of the CDR, it is possible to select a filter with higher selectivity to attenuate high-frequency contribution. In this case, a 1st order filter is considered. The filter is implemented like a proportional and integral path and modelled with two coefficients K_1, and K_2 as shown in Figure 2.7, where K_1 corresponds to the proportional gain, and K_2 represents the integral gain.

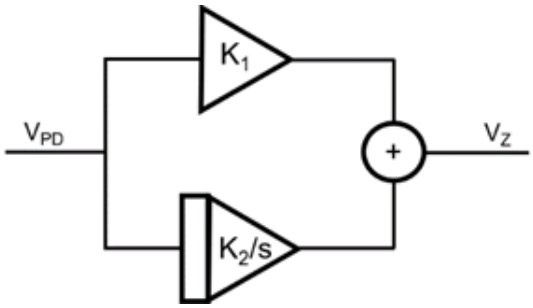

Figure 2.7: Block diagram representation of a 1st-order filter.

The 1st order filter is modelled with the following transfer function:

$$Z_1(s) = K_1 + \frac{K_2}{s} \quad (2.13)$$

When implementing a 1st order filter, such as $Z_1(s)$, the closed loop PLL is a 2nd order system with a transfer function, TF_{CL}, written as:

$$TF_{CL}(s) = \frac{TF_{OL}(s)}{1+TF_{OL}(s)\cdot H(s)} = \frac{\frac{K_{PD}\cdot K_{DCO}}{s}\cdot Z_1(s)}{1+\frac{K_{PD}\cdot K_{DCO}}{s}\cdot Z_1(s)} = \frac{K_{PD}\cdot K_{DCO}\cdot(K_1\cdot s+ K_2)}{s^2 + s\cdot K_{PD}\cdot K_{DCO}\cdot K_1 + K_{PD}\cdot K_{DCO}\cdot K_2} \quad (2.14)$$

Assuming that H(s) = 1/N is the contribution of the frequency divider in the feedback loop system. As known in the literature, a 2nd order system can be expressed in terms of two parameters:

- ω_n = natural frequency.
- ζ = damping factor.

$$TF_{CL}(s) = \frac{2\cdot\zeta\cdot\omega_n\cdot s+ \omega_n{}^2}{s^2+2\cdot\zeta\cdot\omega_n\cdot s+ \omega_n{}^2} \quad (2.15)$$

By visualizing the transfer functions (Figure 2.8) described in Equations 2.6-2.8, it is possible to study the behaviour of the system. In particular, the blue line represents the transfer function of the 1st order low pass filter, the purple line is the open loop transfer function, the green line is the DCO transfer function and the yellow one is the closed loop transfer function.

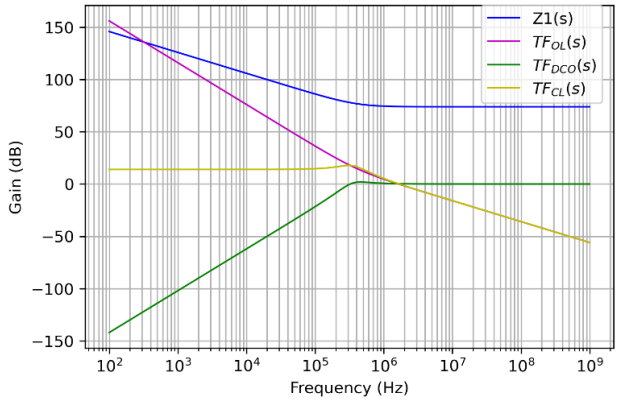

Figure 2.8: The gain of the individual transfer function within a PLL with a 1st-order filter.

The cutoff frequency of TF_{CL} and TF_{DCO} correspond to the ω_n of the system, for instance ≈300kHz in Figure 2.8, therefore, as shown in Equation 2.15-2.17 by changing the filter coefficients (K_1, K_2) it is possible to shift the ω_n and consequently prioritise the filtering of the noise at a higher or lower frequency.

$$\zeta = \frac{K_1}{2}\sqrt{\frac{K_{PD}\cdot K_{DCO}}{K_2}} \quad (2.16)$$

$$\omega_n = \sqrt{K_{PD}\cdot K_{DCO}\cdot K_2} \quad (2.17)$$

By looking at the transfer functions, the noise generated in the DCO is suppressed at low frequencies (to the left side of the ω_n) while the noise introduced by the reference is attenuated at higher frequencies (to the right side of the ω_n).

Once defined the transfer functions, it is possible to extend the analysis to find the system parameters that will minimize the noise contribution from the DCO and the reference and reduce the jitter. The integration range for jitter calculation is [100 Hz - 1 GHz]. By considering a fixed ζ, (e.g, ζ = 1) the optimal ω_n to minimize the jitter is obtained ($\omega_n \geq 100$ MHz) as shown in Figure 2.9. It is possible to extend the analysis further such that also an optimal ζ can be chosen. Considering Figure 2.10, by sweeping the ζ in the range [0.5-5] it is possible to choose an optimal case for which a larger damping factor (e.g., $\zeta \geq 3$) provides a smaller jitter.

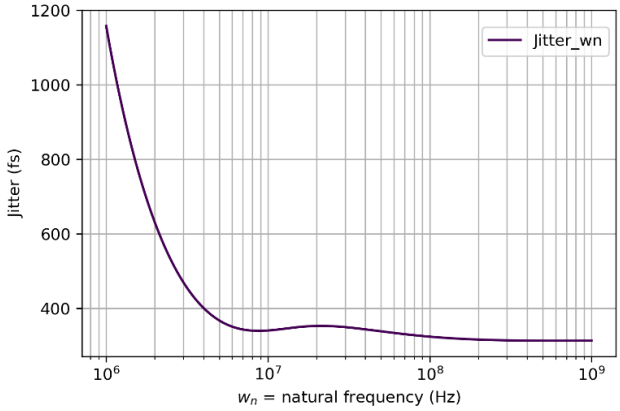

Figure 2.9: Parametric analysis of ω_n to estimate minimum jitter conditions.

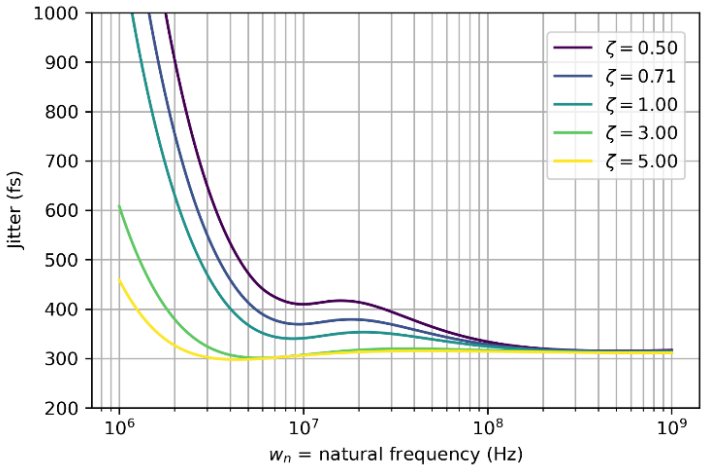

Figure 2.10: Parametric analysis of ω_n and ζ to estimate minimum jitter conditions.

The same analysis is performed considering a 2nd order filter for which a 3rd order system is generated (Appendix A). As a result of the analysis, the choice of a more selective filter is beneficial to suppress higher frequency noise when the reference noise is significantly larger than the DCO noise.

The transfer function analysis aims to describe how the noise propagates through the PLL, therefore both the reference noise of a Fibre Cable at 3.2 GHz and the DCO noise contributions are represented in Figure 2.11.

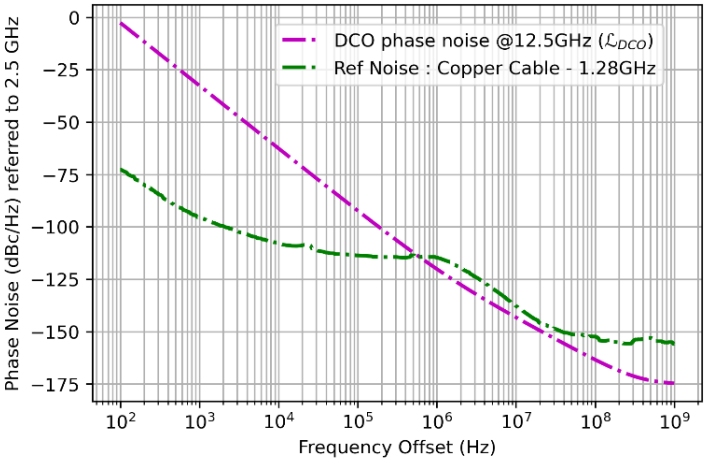

Figure 2.11: Reference noise samples measurement taken at 3.2 GHz using fiber cable and DCO phase noise from DART28 simulations.

By analysing how the noise introduced by the reference data stream and the DCO noise propagate through the PLL it is possible to draw conclusions about the operation of the systems. In particular, it is expected that the reference noise is shaped according to the closed-loop transfer function, as shown in Figure 2.12, while the DCO noise follows the behaviour of the DCO transfer function.

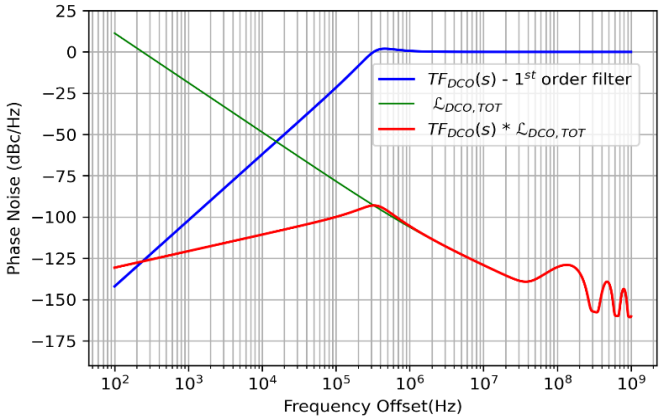

Figure 2.12: DCO phase noise shaped according to the DCO transfer function.

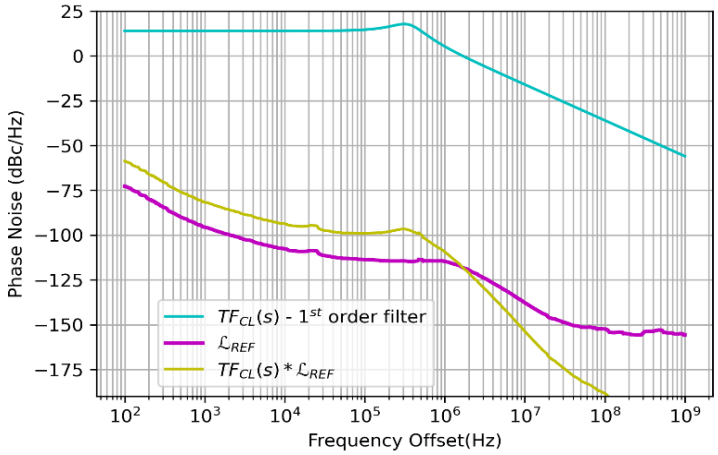

Figure 2.13: Reference noise shaped according to the closed-loop transfer function.

As shown in Figure 2.13, the DCO noise is attenuated at a lower frequency by the DCO transfer function, while the noise introduced at the input of the PLL is attenuated at a higher frequency by the closed-loop transfer function. Choosing a higher order filter provides a higher selectivity beneficial to suppress higher frequency noise when the reference noise is dominating over the other noise contributions.

As a result, the total phase noise in the feedback system is obtained as a combination of the two aforementioned transfer functions, shown in Figure 2.14, such that:

$$\mathscr{L}_{TOT1}(s) = TF_{DCO}(s) \cdot \mathscr{L}_{DCO,TOT} + TF_{CL}(s) \cdot \mathscr{L}_{REF} \quad (2.18)$$

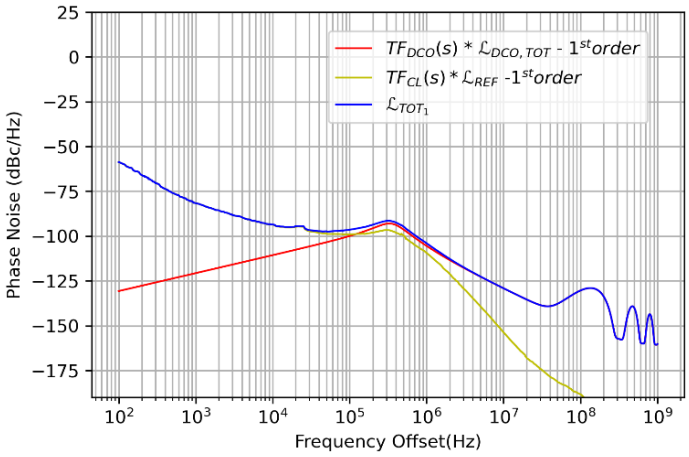

Figure 2.14: Total phase noise with 1st order filter arising from the contributions of the DCO and the reference phase noise.

2.7 Conclusion

This Chapter described a preliminary analysis of the proposed PLL structure, modelled as a 2nd order system using a linear time-invariant system approach. The derived TFs are used to estimate the phase noise contributions from the reference data stream and the DCO in closed-loop PLL operation. From this analysis, it was possible to study how the noise propagates through the PLL and how modelling the feedback system is able to change the dynamic of the PLL. By understanding the behaviour of the oscillator noise is possible to understand how to design the sub-blocks.

For the proposed PLL architecture a 1st order filter has been chosen, for which the coefficients K_1 and K_2 need to be properly defined to generate a proper natural frequency ω_n and damping factor ζ.

Moreover, the advantages and drawbacks of using a 2nd order ΣΔM to drive the DCO are considered. In particular, the trade-off between the order of the ΣΔM, the operation frequency, the power consumption and the PLL bandwidth are taken into account.

As a result of this analysis, the optimal PLL architecture has been chosen. A detailed description of the architecture model and the behavioural implementation is presented in Chapter 3.

Chapter 3 – PLL Architecture Level Description

3.1 Introduction

In this chapter, a brief description of each sub-block followed by an architecture-level description is given. In particular, the Phase Detector (PD) is one of the key blocks since compares the phase of the generated clock and the phase of the reference clock at the input. The oscillation frequency of the DCO is adjusted to be faster or slower based on the output of the PD. In fact, it is the phase difference estimated by the PD that closes the feedback loop. As a consequence of the PD decision, a control word is used to drive the Digital Control Oscillator (DCO) which generates the clock signals at the desired frequency [28]. On top of this, a few other blocks play an important role in the CDR structure. Depending on the applications and the requirements for which the architecture is targeted, the architecture can be optimized to optimize the operation according to its constraints.

In this chapter, the architecture of the proposed PLL-based CDR is introduced. An all-digital PLL-based CDR architecture was chosen for this work because it shows a lower sensitivity to corner and process variations compared to the conventional PLL thanks to the use of a non-linear PD and the digital nature of the architecture. The proposed CDR finds application in a high-speed serial link, that takes as input a data stream with a data rate of 2.5 Gbps and synthetizes an output frequency targeted to be 12.5 GHz. Such a CDR structure is quite similar to a PLL since the main operating principle is analogous, however, it differs only in a few key aspects such as the fact that in a PLL the reference signal is a clock signal with fixed rising/falling edges, while the CDR appeal to rely on a data stream to recover the clock embedded within. One of the pitfalls of this structure is consecutive bits with the same value, resulting in an absence of transitions input signal, which generates an absence of information when no transition is happening at the PD which leads to mistaken decisions. In fact, the CDR relies on the fact that the data stream at the input is continuously switching and an extended absence of transitions due to a constant zero or one signal does not occur. Specific techniques, such as interleaving, may be used to ensure that the input signal will keep on switching even if the data has a low transition density.

3.2 PLL-based CDR Architecture

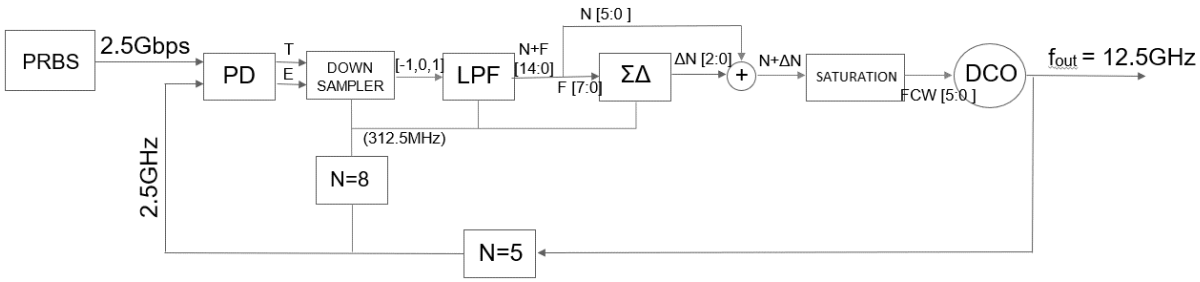

Figure 3.1: Detailed block diagram of the proposed PLL.

A block diagram representation of the proposed PLL-based CDR is presented in Figure 3.1, in particular, the building blocks are the following:

- Alexander Phase Detector (PD),
- Downsampler,
- Digital Low Pass Filter (LPF),
- 2nd order ΣΔM (MASH 1-1),

- Digital Control Oscillator (DCO), and
- Integer Frequency Divider (N).

In addition, the PRBS block shown in Figure 3.1, which is part of the testbench, is a Pseudo-Random Bit Sequence Generator that is the base block to introduce random input data from which the targeted clock signal is generated.

3.3 BBPD - Alexander Phase Detector

The proposed Alexander PD, also found in literature as a Bang-Bang PD is, shown in Figure 3.2. The PD compares the phase of the input data and the recovered clock and provides information to correct the oscillation frequency of the oscillator. This PD is non-linear since it only provides information about the sign of the phase of the two signals but does not give any information about the magnitude of the phase error. Indeed, the generated output gives information detecting if a transition occurred and if the input data come earlier or later w.r.t. the reference clock, but no information to quantify the phase difference is given. Its operation can be understood by observing three consecutive samples verifying when a transition occurs at the output. A detailed description of this system is given in [16], [17], [18],[19], [22], [29].

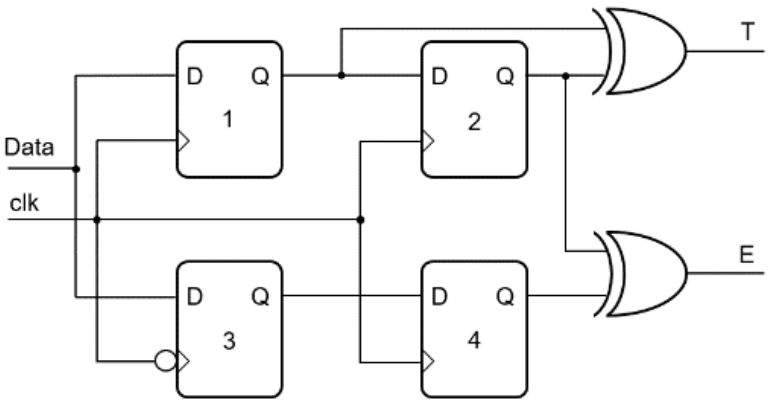

Figure 3.2: Gate-level representation of the Alexander PD.

The PD provides information in terms of Transitions, with T=1 when it occurs and as Early, with E=1 when the data is earlier than the clock signal. The operation can be modelled as:

$$T = Q_1 \oplus Q_2 \quad (2.1)$$

$$E = Q_2 \oplus Q_4 \quad (2.2)$$

where Q_1, Q_2, Q_3, and Q_4 are respectively the output of the D-flip flops referred to with the same number. It is important to clarify that when T=0, the information E=1 or E=0 is meaningless because if no transition occurs, no valid phase information can be extracted.

T	E
1	0
1	1
0	1
0	0

Table 3.1: Summary of the output of the PD.

The operation of the PD can be modelled as a state machine defined in three states: Early, Late or no commutation.

The basic design idea is that for each clock period, a decision to adjust the output frequency of the DCO is required. As a result, there are three possible conditions represented:

- The DCO should be increased if the output of the PD is T=1, E= 1,
- The DCO should be decreased if the output of the PD is T=1, E= 0,
- The DCO should not change the configuration if the output of the PD is T=0, E= 0, or E= 1.

One of the most important parameters of a phase detector is its gain (K_{PD}), which can be calculated by considering the Cumulative Distribution Function (CDF), shown in Figure 3.4. The CDF indeed represents the probability that a transition occurred, and the output is Early w.r.t. the clock, (P(E=1)), and is defined by the following equation, [20]:

$$\text{CDF} = \frac{1}{2}\left(1 + erf\left(\frac{t-\mu}{\sqrt{2}\sigma}\right)\right) \quad (2.3)$$

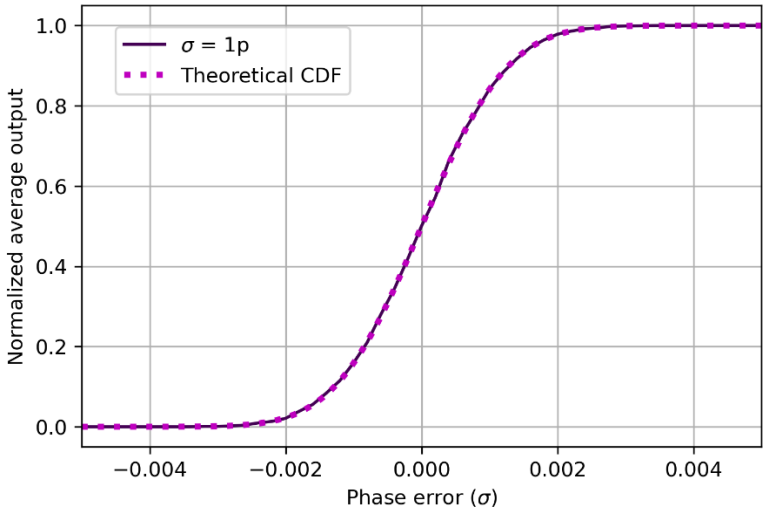

Figure 3.4: Error function of the PD obtained using a reference jitter of σ=1ps.

The phase detector gain is estimated from the error function as follows:

$$K_{PD} = \max\left(\frac{\frac{d(CDF(\sigma)}{dt}}{2\cdot\pi\cdot f_{ref}}\right) \quad (2.4)$$

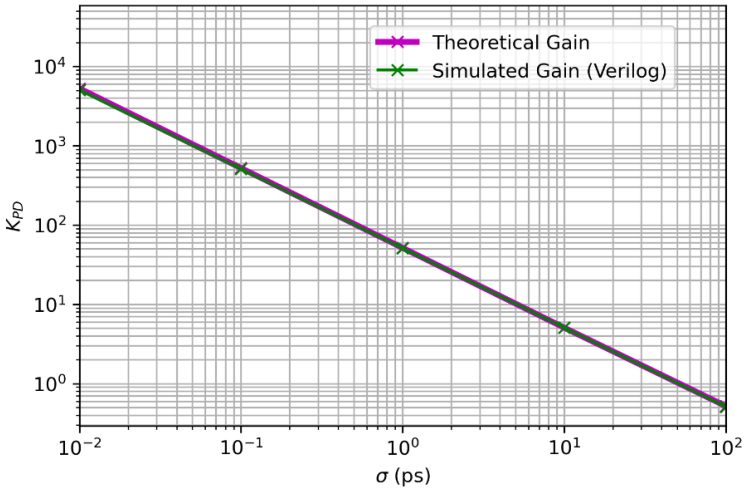

Figure 3.5: PD gain estimation obtained considering a jitter (σ) = [10fs-100ps].

As shown in Figure 3.5, the gain of the PD is strongly dependent on the jitter present at the input of the PD. A larger jitter causes a reduction of PD gain, while small jitter increases the detection gain.

3.4 Downsampler

The block following the PD in the proposed topology is the downsampler, which is used to reduce the rate of phase information to be processed by the loop filter. The PD works at the speed of the input data (2.5 GHz), while the output data of the downsampler is targeted to be generated at 312.5 MHz. The operation speed is reduced with a ratio (8:1) w.r.t. the input data rate. A reduced operation in digital logic is preferred since it reduces the power consumption of the digital logic and relaxes its timing requirements. On the other hand, this comes at the cost of a limited and approximated PD decision. This downsampler presents the simple version, which takes only the first value generated by the PD out of 8 decisions. In this case, the first decision drives the DCO, while the successive 7 samples are discarded. This results in a non-optimal solution, since the discarded information (7/8) can represent a completely different decision compared to the first decision.

Additionally, the downsampler converts the output of the PD into one of three possible values such as (-1, 0, 1) suitable for use by the loop filter using the convention shown in Table 3.2.

T	E	DATA
0	0	0
0	1	0
1	0	1
1	1	-1

Table 3.2: Conversion table of the PD generated by the downsampler.

In the next chapter, three downsampler topologies will be compared.

3.5 First-Order Digital Loop Filter

The downsampler output is then processed by the digital loop filter. According to the preliminary analysis presented in Chapter 2, the architecture requires a 1st order low-pass filter.

In this section, the block diagram is shown. The filter is constituted by a proportional-integral (PI) control loop, shown in Figure 2.7, for which the proportional gain is defined as K_1, and the integral gain is K_2. By adjusting these two parameters it is possible to change the bandwidth and damping of the 2nd order PLL. The filter is a digital system, in which the units are referred to the number of bits used to represent the input/output signals and are expressed in bits.

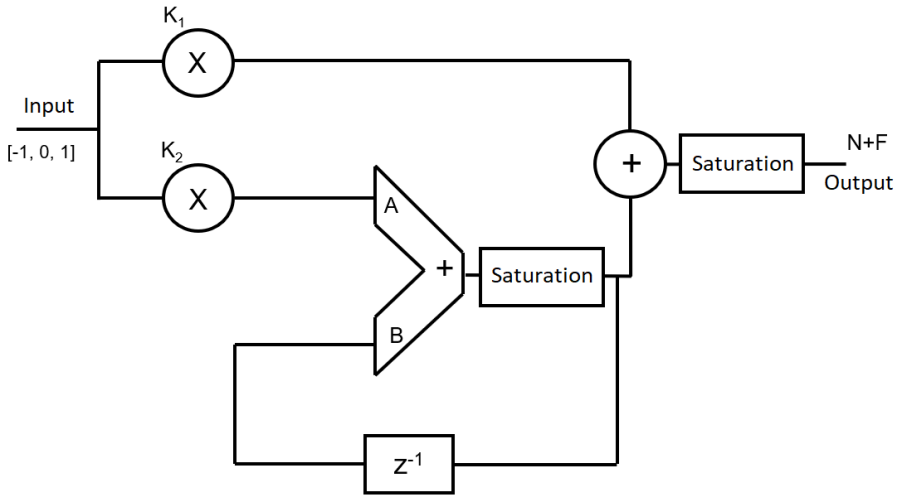

Figure 3.6: Block diagram of a 1st order Digital Filter.

A block diagram of the 1st order Digital Filter is shown in Figure 3.6. The *Input* bus can assume values such as [-1, 0, 1] and is processed in the proportional and integral paths. From a calculation point of view, the proportional path performs a multiplication of *Input* and K_1, while the integral path takes the *Input* data and multiplies it by a coefficient K_2 and the result is fed into an integrator. The integrator is implemented using an adder and a D flip-flop. The result of the addition is fed back into the adder to implement the integration and the outcome of the two paths is then summed. The use of a limiter indicated as Saturation in Figure 3.6, along the path is beneficial for digital calculation purposes. The limiter is indeed required to prevent the result of the sum from "rolling over" from large positive values to large negative values, due to the limited size of the register. The saturation logic has the purpose to clamp to the maximum value attainable and prevent overflow conditions. The *Output* bus generated in the filter is then split into two consecutive sequences considered as an integer (N) and a fractional (F) part. The fractional part is then further processed in the Multi-stAge noise SHaping (MASH) ΣΔM (ΔF) and recombined subsequently with the integer output (N).

3.6 ΣΔ Modulator - MASH 1-1

As mentioned in Chapter 2, the use of a 2nd order ΣΔM can be beneficial to reduce the quantization noise introduced by the DCO tuning word. When connected to the input of the DCO a ΣΔM can indeed reduce the phase noise since it moves the quantization noise to a higher frequency with a slope of +20 dB/dec per order, [21], [22]. Subsequently, the noise is filtered by the DCO. The ΣΔM is used to implement a digital processing technique that dithers the fractional part of the signal generated by the filter such that

the average of the output generated is the same as the input. In particular, the ΣΔM topology chosen is a 2nd MASH (Multi-stAge noise SHaping) ΣΔM.

From the analysis shown in Chapter 2, the optimal configuration to minimize jitter ($L_{DCO,\,TOT}$) is represented by a 2nd order ΣΔM operating at 312.5 MHz ($f_{\Sigma\Delta M}$).

From an architectural point of view, the fractional part of the signal generated by the 1st-order filter is further processed by a ΣΔM. In particular, the data generated by the filter is in the format of 14-bit strings, with an integer pattern of 6 bits for the integer part and 8 bits for the fractional part. In the end, the output generated by the MASH 1-1, (ΔF) is then combined with the integer data generated by the filter, (N), to get the frequency control word used to drive the DCO.

The operation principle of a ΣΔM is based on two properties: oversampling and quantization error shaping. The idea of Oversampling is strictly bounded to the Nyquist theorem, for which the minimum sampling frequency (f_s) needs to be twice the signal bandwidth, i.e., $f_s=2\cdot Bw$. Under this assumption, Nyquist frequency f_N is equal to f_s. However, when the oversampling is considered, $f_s >> f_N$, and the ratio is defined as Oversampling Ratio (OSR).

The noise shaping shifts most of the quantization noise to higher frequencies which gets filtered right after. Therefore, by choosing the oversampling frequency and the order of the ΣΔM it is possible to shape the quantization noise, such that the bandwidth of interest is selected and the signal outside the bandwidth of interest is attenuated. The quantization noise generated is expected to have a slope of +20 dB/dec per each order of the ΣΔM. In this case, is +40 dB/dec since it is of 2nd order, as shown in Figure 3.7.

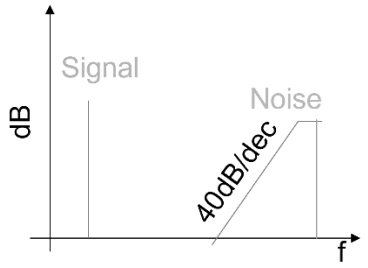

Figure 3.7: Expected Output behaviour of 2nd order ΣΔM.

The proposed structure, shown in Figure 3.8, is a 2nd order ΣΔM which is constituted by a Noise Cancelling Logic (NCL), shown in Figure 3.10, and two cascaded 1st order ΣΔMs obtained by using Error Feedback Modulators (EFM) as shown in Figure 3.9. In particular, the ΣΔM has the purpose to introduce dithering in the DCO control word to limit the generation of spurious tones.

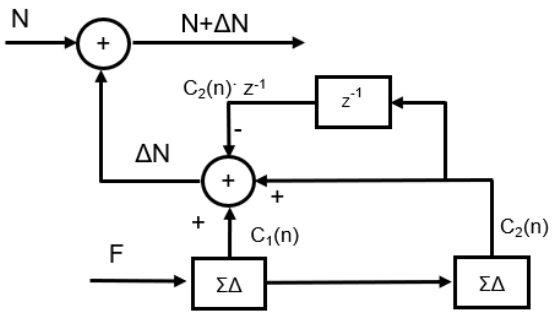

Figure 3.8: Block implementation of a ΣΔM - MASH 1-1.

3.6.1 Error Feedback Modulator

A single stage 1st order ΣΔM, also called Error Feedback Modulator (EFM), is one of the main components of the MASH. The EFM, shown in Figure 3.9, aims to perform some important operations such as:

- Takes the input sequence $X_i[n]$, which is represented by the fractional data (F) generated by the LPF. The EFM is constituted by an adder which accumulates the input A and the input B which is the result of the integration of the sum. The operation of the accumulator is performed until the result overflows. The carry (c_n) is the information required to perform the noise shaping.
- A D-flip flop combined with the accumulator works as an integrator that shapes the noise with a high-pass response behaviour.

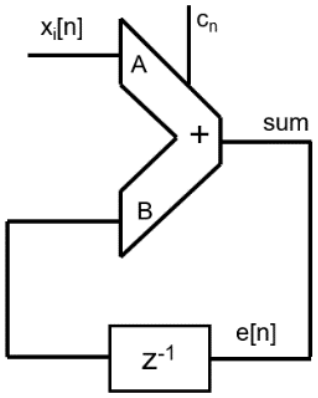

Figure 3.9: Single-stage implementation of the Error Feedback Modulator.

The behaviour of a single stage can be modelled as:

$$C_1(n) = e_1(n) \cdot (1 - z^{-1}) + X_i(n) \quad (2.5)$$

The implementation is referred to a discrete system for which n represents the iterative step. According to Equation 2.5, the signal $C_1(n)$ is composed of the input signal and the quantization noise shaped with a high-pass shape. Furthermore, the result of the sum is then passed to a second stage EFM which processes the information generating another carry.

Subsequently, the carry $C_1(n)$ is further processed in the Noise Cancelling Logic (NCL), while the sum is processed further by the second stage EFM. The 2nd stage is based on the same operation principle and the carry obtained is then elaborated in the Noise Cancelling Logic as well.

3.6.2 Noise Cancelling Logic

The Noise Cancelling Logic (NCL), shown in Figure 3.10, is able to cancel the quantization noise coming from the previous stage, such that the generated output (ΔN) is the sum of the two carries, one from each stage, with the carry of the second stage integrated as defined by Equation 2.6:

$$\Delta N = C_1 + C_2 - C_2 \cdot z^{-1} \quad \in [-1;\ 2] \quad (2.6)$$

And it is expected ΔN to correspond to the average of the input signal (F). For a second-order modulator, ΔN can assume values within the range of [-1; 2]. An example of the

output of the NCL is given in Figure 3.11. The proposed study case assumes a constant input signal of the ΣΔM.

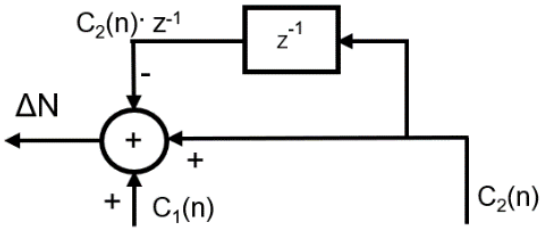

Figure 3.10: Block diagram implementation of the Noise Cancelling Logic.

Combining Equations 2.5 and 2.6 the output of the NCL can be written as:

$$\Delta N[n] = e_2[n]\cdot(1 - z^{-1})^2 + X_i[n] \quad (2.7)$$

Where X_i is the input of the MASH and the quantization noise resulting is the one generated by the second stage and e_2 is the quantization error resulting from the second EFM.

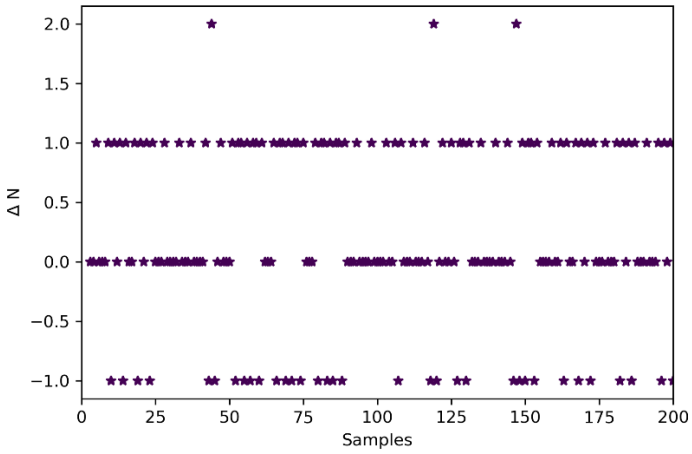

Figure 3.11: Output of the Noise cancelling logic in the time domain. Data taken from simulation.

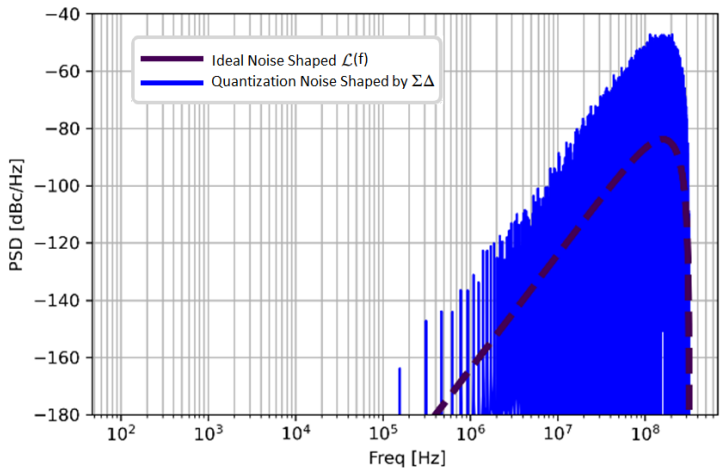

Figure 3.12: Output of the noise cancelling logic in the frequency domain obtained from data in Figure 3.11.

The Power Spectral Density (PSD) of the noise-shaped signal is shown in Figure 3.12 and Equation 2.8 is used to approximate its behaviour as shown with the dashed line, [7]. From Figure 3.12, the two curves do not perfectly overlap since some additional spurious components are caused by the periodicity of the input signal simulated, which increases the PSD contribution. A constant signal indeed generates a periodic pattern in the time domain that translates into spurious harmonics, as shown in (Figure 3.11),

$$\mathcal{L}(f) \propto \frac{1}{12 \cdot f_s} \, 2 \cdot \sin\left(\frac{\pi \cdot f}{f_s}\right)^2 \quad (2.8)$$

3.7 LC Digital Control Oscillator

The DCO is one of the key building blocks of the PLL since it synthesizes the desired clock signal. The output frequency of the DCO is related to the Frequency Control Word (FCW). The FCW is obtained as the sum of the Integer output of the Filter (N) and the output of the ΣΔM (ΔN). The oscillator is implemented as an LC resonant circuit as shown in Figure 3.13. This circuit is constituted by an inductor L and a fixed capacitor C [23]. The frequency of the oscillator can be changed by selecting a different control word, that changes the configuration of the capacitor array.

This thesis is mainly focused on the study of noise propagation on a CDR. For this purpose, only a small tuning range is used and a better understanding of noise effects in the CDR.

According to Equations 2.9 and 2.10, it is possible to size both the capacitors and the inductor such that the resonant circuit oscillates at the targeted frequency (f_{center}=12.5 GHz;). Furthermore, it is possible to control the oscillation frequency in a specific range. In this case, an array of additional binary weighted capacitors properly sized are added and connected through a variable control word properly defined.

$$f_{osc} = \frac{1}{2 \cdot \pi \cdot \sqrt{L \cdot (C_{TOT})}} \quad (2.9)$$

$$C_{TOT} = C + \sum_{N=0}^{5} b_i \cdot 2^N \cdot Cu \, (2.10)$$

where L= 150pH, C = 1.078pF is a fixed capacitor and C_u = 70aF is the unit capacitor value of an array of 64 capacitors.

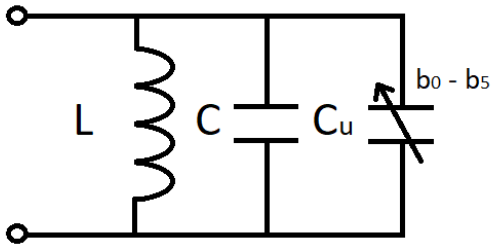

Figure 3.13: Simplified representation of an LC Oscillator.

In particular, the auxiliary capacitor array is constituted by 6 binary weighted capacitors such that a 6-bit control word can be used to change the oscillation frequency. As a result, the DCO is designed such that f_{min}=12.487 GHz, f_{max}=12.512 GHz and a single capacitor can induce a single variation f_{delta}= 404.79 kHz. The tuning range of the DCO is shown in Figure 3.14.

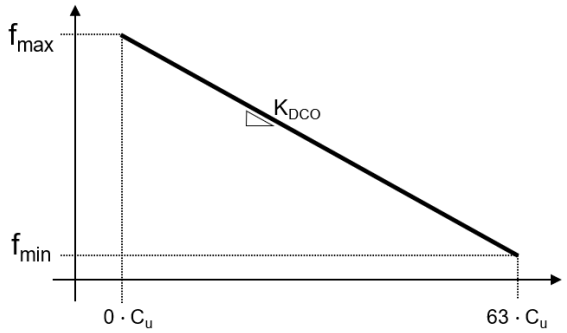

Figure 3.14: Oscillator Frequency versus number of capacitors.

3.7.1 DCO Noise Modelling

The noise contribution of a CDR is one of the main concerns of this work. Is therefore important to understand how the noise propagates through the system. In a CDR, phase noise cannot be completely suppressed, but by knowing the main sources of noise it is possible to mitigate their contribution.

As previously mentioned in Chapter 2, the noise spectrum can be modelled in three regions, as shown in Figure 3.15. The extended Leeson's equation (2.8), [1] models the phase noise of an oscillator using regions of $1/f^3$, $1/f^2$ and noise floor. In particular, the noise floor gives a flat contribution and is dominant at high frequencies, instead, $1/f^2$ has a slope of 20 dB/dec and is the result of the integration of the noise floor when it propagates in the VCO, while the $1/f^3$ has a slope of 30 dB/dec and is the result of the integration of the Flicker noise. In particular, Leeson's equation is useful in transfer function analysis, but cannot be directly used in behavioural simulations. However, it is useful for quantifying the contribution of noise sources and fitting their contribution to a reference noise model of the DCO.

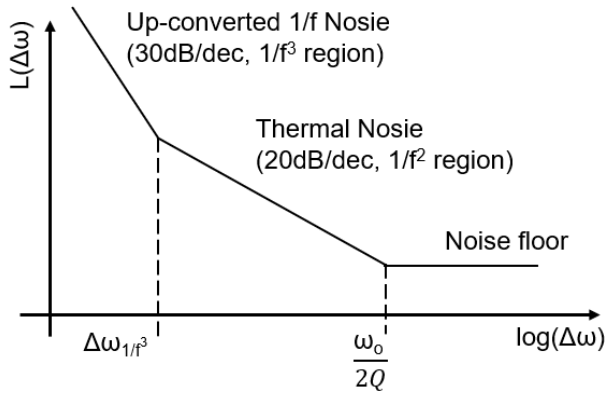

Figure 3.15: Phase noise characterization of the DCO.

In fact, the effects of the phase noise are reproduced in the DCO Verilog model, such that the contribution of period jitter and phase jitter are reproduced c. Due to the phase noise and the jitter, the clock frequency generated by the DCO can suffer from some variations in the clock periodicity or in the phase, making each clock period slightly different from the previous one. These variations in the oscillation frequency of the DCO translate into higher phase noise in the CDR.

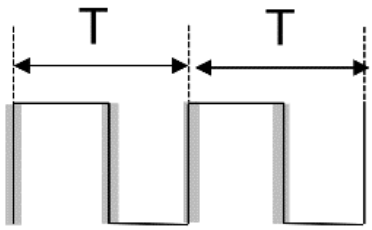

Figure 3.16: Phase Jitter of the DCO, deviate from the ideal edge.

In particular, the phase jitter is the small variation in the rising and falling edge with respect to the ideal case, as shown in Figure 3.16. This variation occurs randomly and the variation on each edge is independent of the others and is smaller compared to period T. This effect is in general due to the local temperature variation in the circuit that causes the oscillator to wander above and below the average period.

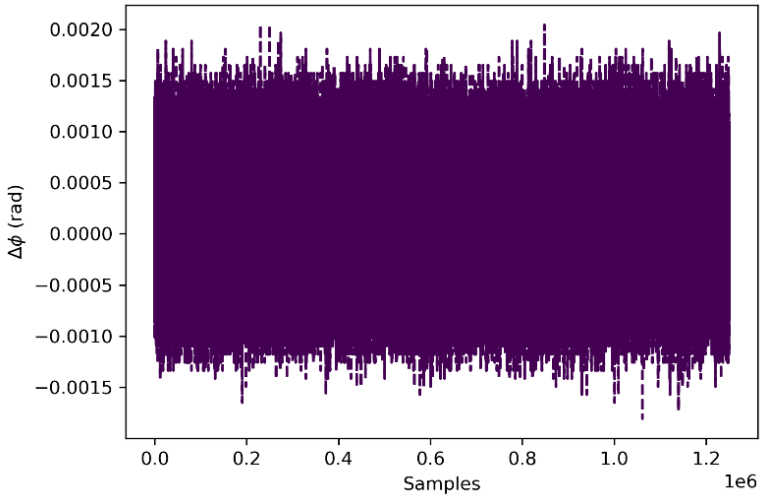

Figure 3.17: Phase error of the DCO in the time domain. Data taken from simulation.

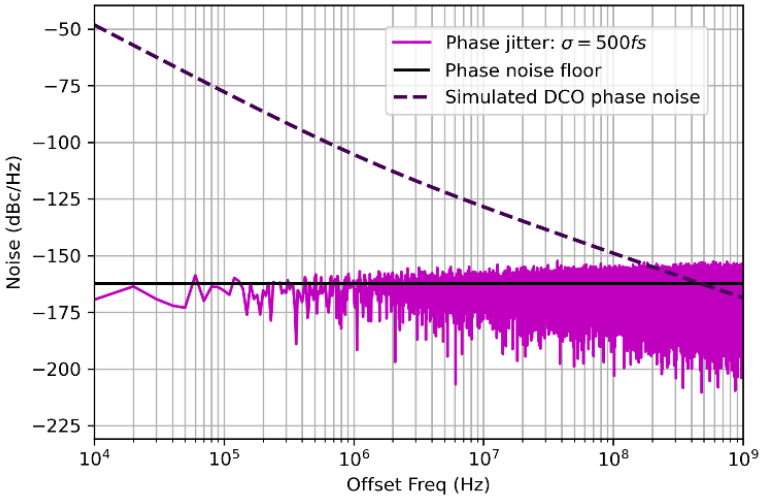

Figure 3.18: PSD of the phase error in the frequency domain obtained with the data in Figure 3.17.

The phase noise in the time domain is then modelled as Gaussian variation on the edge of the clock ideal period. The phase error caused by the phase jitter is then shown in

Figure 3.17. In the frequency domain, the PSD of the signal simulated generates a noise floor contribution of the overall noise generated in a DCO, as shown in Figure 3.18.

A second contribution considered comes from the period jitter, which causes small variations in the period T. Ideally each period should be T, however, a small variation from the ideal period.

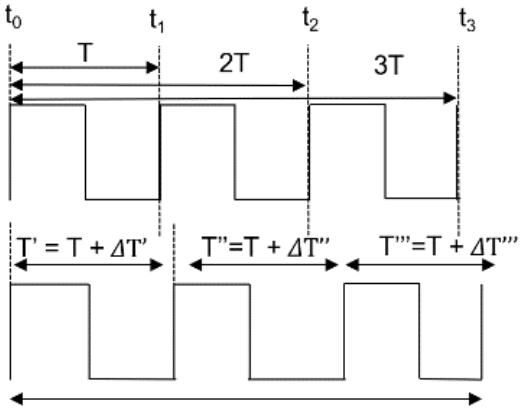

Figure 3.19: Ideal period on top and a realistic case of the clock signal with period jitter at the bottom.

As shown in Figure 3.19, in this case, the period jitter contribution causes a cumulative effect on the collected sample, such that each sample is conditioned by the variation of the previous period. This variation can be modelled as a random walk where each period is conditioned by the variation of the previous and in turn can affect the next period, as shown in Figure 3.20. In the frequency domain, this contribution generates the $1/f^2$ portion of the spectrum, as shown in Figure 3.21.

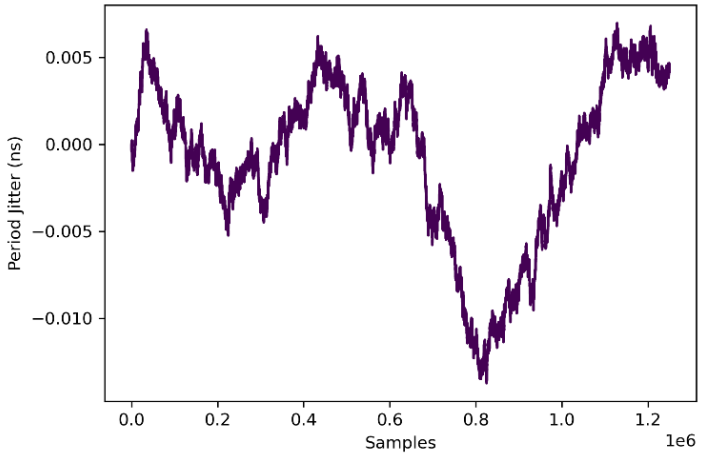

Figure 3.20: Period error of the DCO in the time domain. Data taken from simulation.

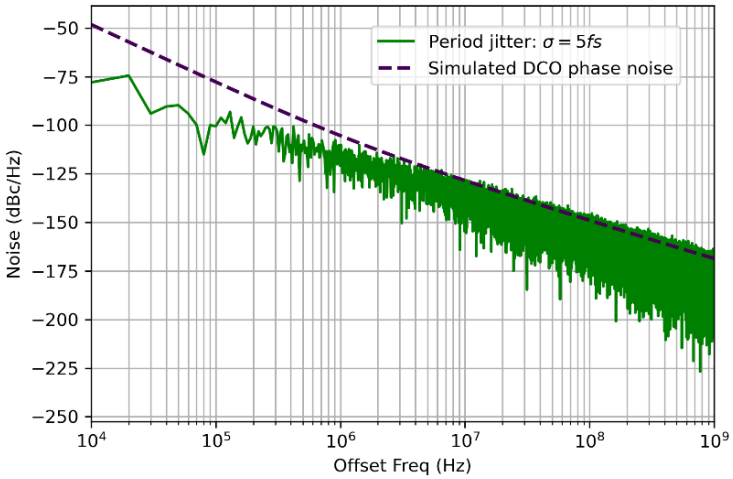

Figure 3.21: PSD of period error in the frequency domain obtained with the data in Figure 3.20.

In this case, the contribution of period jitter and phase jitter represents the main noise influence. To collect the samples for each period, phase jitter and period jitter are measured as:

$$\Delta T = N \cdot T - (T_0 - T_N) \quad (2.11)$$

Where ΔT is the variation from the ideal case, T is the ideal period, N is the number of periods considered, T_0 is the size of the first period measured and T_N is the size of the N period. The linear sum of the two noise contributions gives the overall contribution. Ans is shown in Figure 3.22. The sigma in the noise variation is chosen such that the simulated noise fit with the data collected from the DART 28 simulation, [6].

It is assumed that the $1/f^3$ contribution is not relevant in this case. This assumption is validated by the calculation of the jitter of both the simulated DCO noise from DART28 and the Verilog model of the DCO.

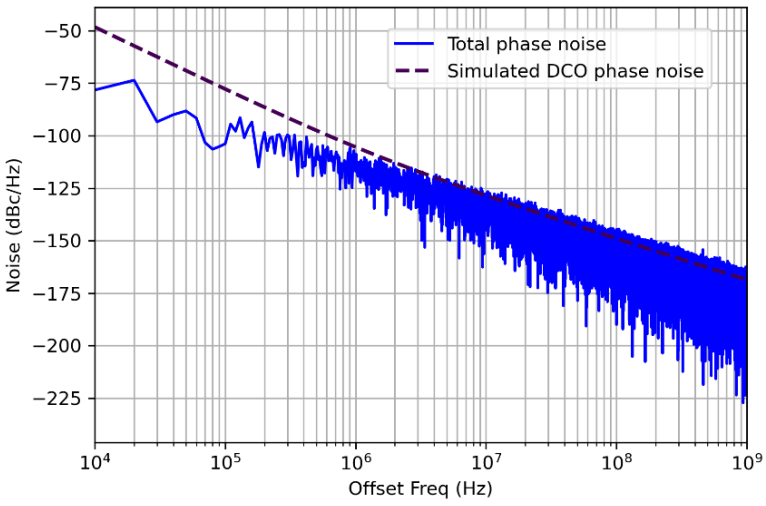

Figure 3.22: Total noise contribution obtained as a combination of phase jitter and period jitter previously calculated in Figures 3.18 and 3.21.

3.8 Frequency Divider

The last block presented is a frequency divider which closes the feedback loop. This block takes the signal synthesized by the DCO at 12.5 GHz as input and generates an output of 2.5 GHz. which is 5 times slower.

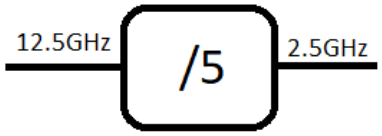

Figure 3.23: Block diagram of the frequency divider.

Multiple dividers with different division ratios may be used within a CDR, such that from an initial clock signal several different internal clocks are generated. For instance, a second frequency divider in this CDR is used to trigger the operation of the LPF and the ΣΔM, which operate at 312.5 MHz. The division ratio desired, in this case, is 8 considering as input the 2.5 GHz clock generated by the first clock divider. The representation of the frequency dividers is shown in Figure 3.1.

3.9 CDR Testbench

To verify and characterize the operation of the CDR a generic testbench is shown in Figure 3.24. When the loop is locked, it is expected that the output frequency generated is 12.5 GHz. To measure the phase error, the generated output frequency is compared to an ideal 12.5 GHz input signal.

The input clock signal at 12.5 GHz is used also as a trigger to generate random input data bits. In particular, the ideal 12.5 GHz signal is divided by 5 using a frequency divider to generate a 2.5 GHz clock signal. This clock is fed into a variable delay that is used to synthesize the input jitter of the CDR. A PRBS generator is used to create random data at the input of the CDR to test the operation of a CDR. The output produced by the CDR is then processed to calculate the phase variation and to calculate PSD, FFT, and jitter variance or to analyse its transient behaviour.

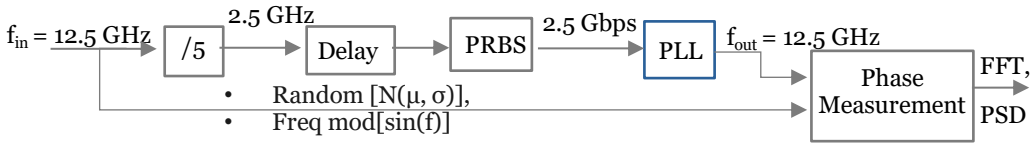

Figure 3.24: Block diagram representation of the Testbench used to evaluate the performance of the CDR.

3.10 CDR Operation

In this section, the dynamic of the implemented CDR model is presented. In Figure 3.25 the settling time of the CDR is shown. After an initial locking transient, the signal generated converges to the expected center frequency of 12.5 GHz. In the shown scenario, the time to lock is about 10.4μs calculated as T=80ps as the period at 12.5 GHz and about 130000 samples.

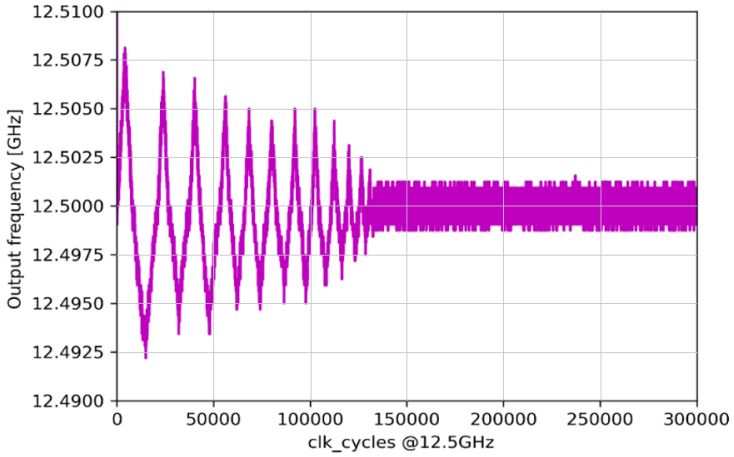

Figure 3.25: Example of the operation of the CDR in the time domain. Time to lock ≈ 80ps * 130000 ≈ 10.4us.

The settling time of the CDR can be reduced by changing the filter coefficients to proportional and integral (K_1 and K_2 respectively). For instance, it is possible to reduce the time to lock. Two possible configurations are shown in Figures 3.25 and 3.26.

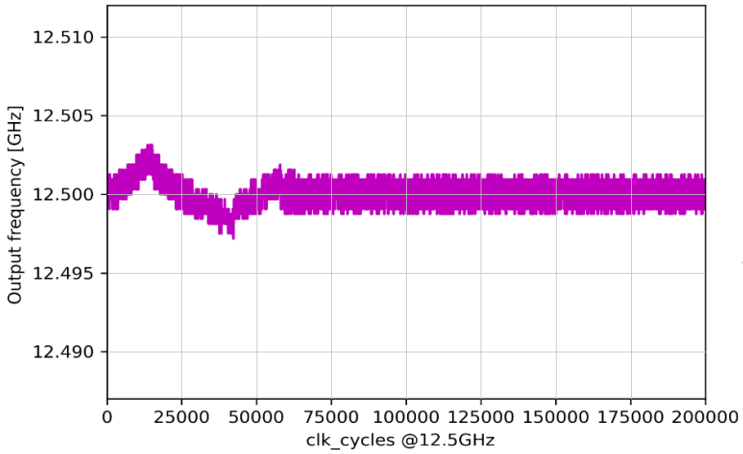

Figure 3.26: Example of the operation of the CDR in the time domain. Time to lock ≈ 80ps * 70000 ≈ 5.6us.

The operation principle of the CDR is to reduce the phase error between the data rate and the output by changing the oscillation frequency of the DCO. However, the tuning range of a CDR is limited by the tuning range of the DCO. As a result, to correct the phase error, the CDR can modify the output frequency in the convergence range, as shown in Figures 3.27, and 3.28. To verify that the CDR is working properly, the frequency of the input signal is slightly changed to higher or lower values. As a result, it is expected that the output frequency follows the input variation. In particular, for this design the tuning range of the DCO is chosen to be ±0.1% of the center frequency, therefore it is expected to generate a frequency shift up to ±0.1%, which corresponds to a tuning range (12.487 GHz - 12.512 GHz) with a frequency resolution of 404kHz.

Frequency shift	Expected output
+0.1%	12.5125 GHz
-0.1%	12.4875 GHz

Table 3.3: Expected output Frequency shift = ±0.1% in the time domain.

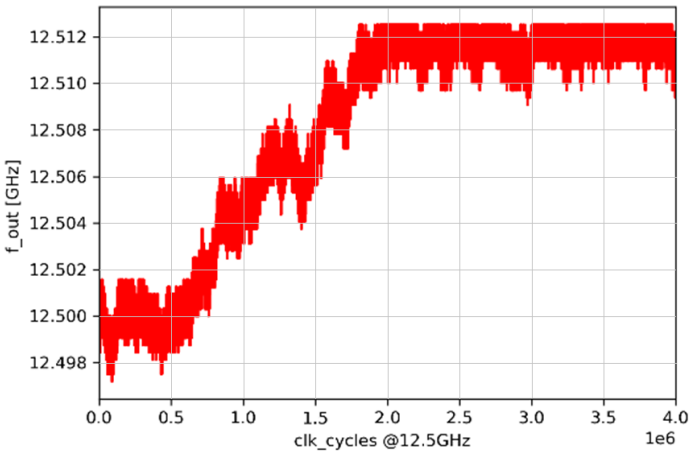

Figure 3.27: Simulation results for CDR with expected output frequency shift = +0.1% in the time domain.

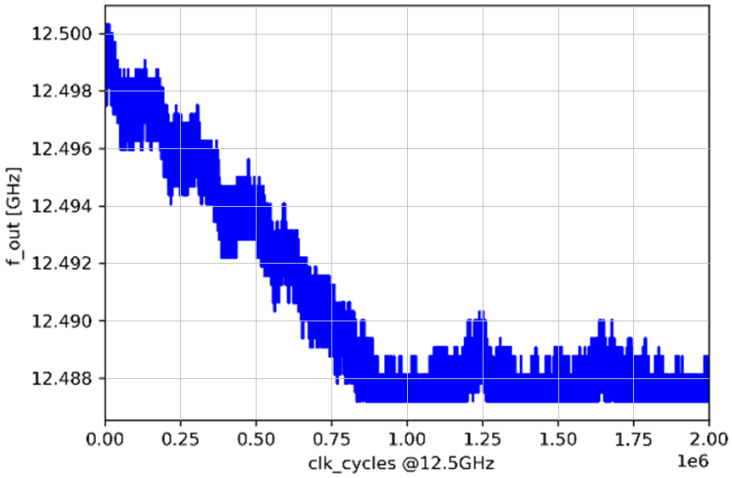

Figure 3.28: Simulation results for CDR with expected output frequency shift =-0.1% in the time domain.

The locking range of this CDR is chosen rather small on purpose to focus the analysis mainly on the consequences of the phase noise on the CDR.

The proposed CDR can correct the output frequency with steps of 404kHz in the expected range. However, if the frequency shift required is higher than the locking range, as shown in Figures 3.29, and 3.30, the CDR is not able to lock. Indeed, if a larger output frequency range is targeted, the CDR tries to correct the phase error detected by increasing or decreasing the DCO frequency. At some point, the phase error accumulates after many periods and translates into a random variation in the output frequency generated.

In this case, an output frequency outside the locking range is targeted at (±0.5%), which corresponds to frequencies of 12.437 GHz - 12.562 GHz, which cannot be attained considering the current DCO design, making it impossible for the CDR to lock.

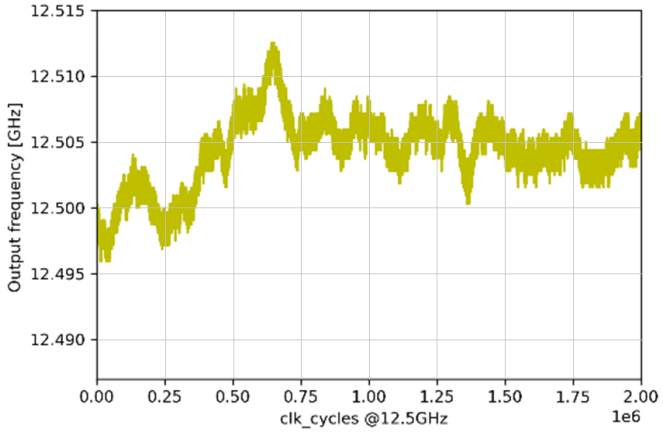

Figure 3.29: Simulation results for CDR with expected output frequency shift with shift = +0.5% which is out of the locking range in the time domain.

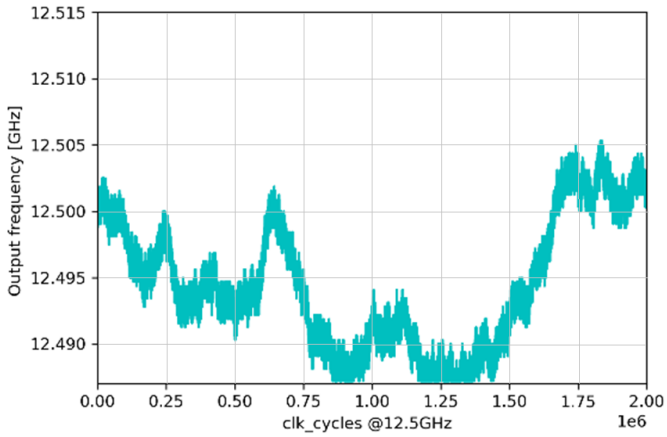

Figure 3.30: Simulation results for CDR with expected output frequency shift = -0.5% which is out of the locking range in the time domain.

Frequency shift	Expected output
+0.5%	12.5625 GHz
-0.5%	12.4375 GHz

Table 3.4: Expected output frequency shift = ±0.5% in the time domain.

It is also important to understand how changing the filter coefficients can affect the performance of the CDR in the frequency domain. By changing K_1, and K_2 in the filter it is possible the change the transfer function. According to the analysis presented in Chapter 2, changing the natural frequency of the 2nd order filter corresponds also to making the feedback system more selective to reference noise or to the DCO noise. Therefore, the system behaviour can be adapted to balance the noise contribution between the reference and DCO phase.

The filter coefficients are constituted by 14 unsigned bits whereof 6 integers and 8 fractional. An example is shown below, for which the value of the proportional and integral is given both in binary and in decimal.

- K_1 = 14'b000000_11110101 = 0.95703125
- K_2 = 14'b000000_00001001 = 0.03515625

By estimating the PSD of the phase error, the CDR output phase spectrum can be obtained. From this is possible to estimate ω_n and the ζ of the feedback system. For example, the noise spectrum obtained considering the filter coefficients previously mentioned is shown in Figure 3.31. In this case, the CDR has a natural frequency (ω_n) at 6-7 MHz.

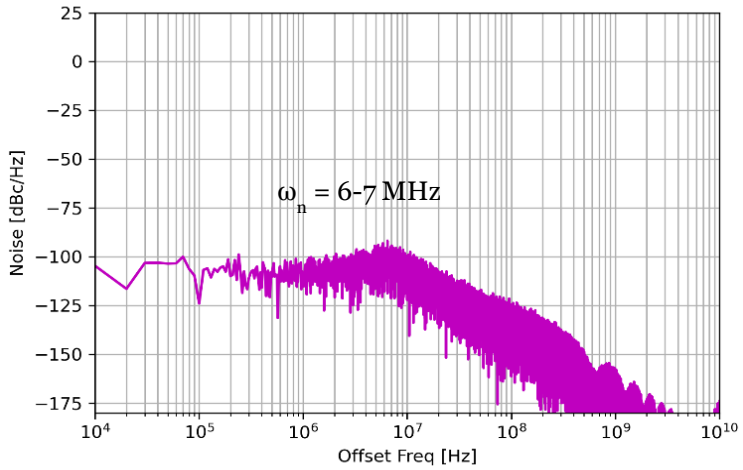

Figure 3.31: Phase noise of the CDR with ω_n = 6-7 MHz. Data taken from simulation.

By selecting different parameters, is it also visible that the bandwidth of the CDR is also shifted, as shown in Figure 3.32. This can be advantageous to reduce the jitter. By moving the ω_n to the right, the noise coming from the reference is further attenuated, while moving the ω_n to the left a filtering on the DCO noise is applied.

- K_1 =14'b000000_00010101 = 0.04296875
- K_2 = 14'b000000_00001001 = 0.03515625

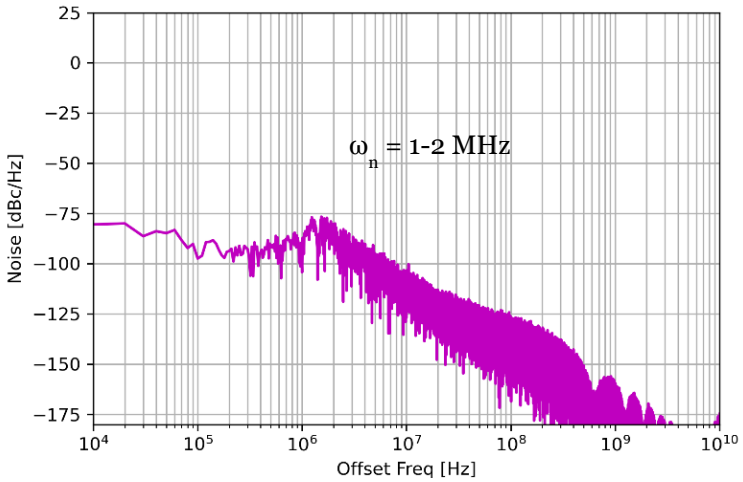

Figure 3.32: Phase noise of the CDR with ω_n = 1-2 MHz. Data taken from simulation.

As a rule of thumb, according to Equations 2.16 and 2.17, it is possible to conclude that:

- Larger K_1 causes ω_n to shit right and increases ζ.
- Larger K_2 causes ω_n to shift left and reduces ζ.

These parameters need to be chosen as a trade-off considering the stability of the loop, its transfer function as well as its transient response which defines the time required to lock the CDR.

3.11 Noise Contribution on the PSD

In this section, several sources of noise are discussed. The performance is evaluated by calculating the jitter from the PSD of the output frequency. Welch method, [30], is also used to compare the results obtained. The phase noise of the CDR with no external noise source is given in Figure 3.33. The phase noise results from the PD noise and the phase error produced at the output of the CDR.

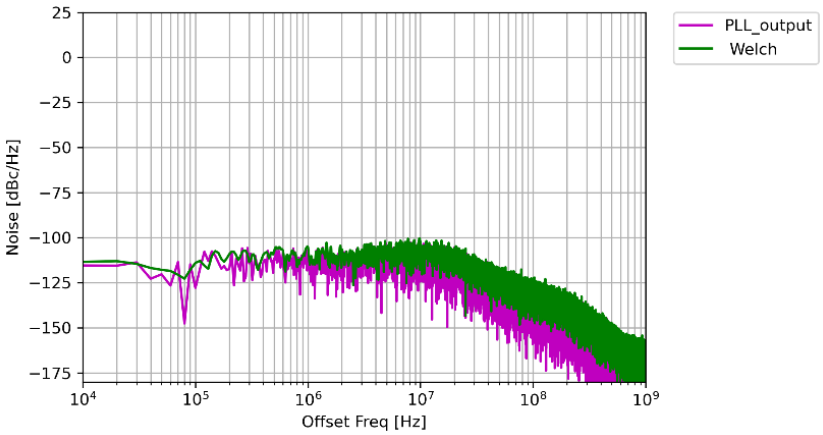

Figure 3.33: Phase noise of the CDR with no external noise source applied to the system. The jitter contribution when only quantization noise is considered is 265.88fs.

As already mentioned, the DCO noise can be split into two main contributions: phase jitter and period jitter. In particular, the phase jitter is mostly visible at higher frequencies as a noise floor contribution. For this purpose, the noise contributions coming from the DCO are separately considered. By comparing two cases (phase jitter = 100fs) in Figure 3.34 and (phase jitter = 500fs) in Figure 3.35 is it visible, also highlighted with the yellow circle, that the contribution can be significant for larger sigma.

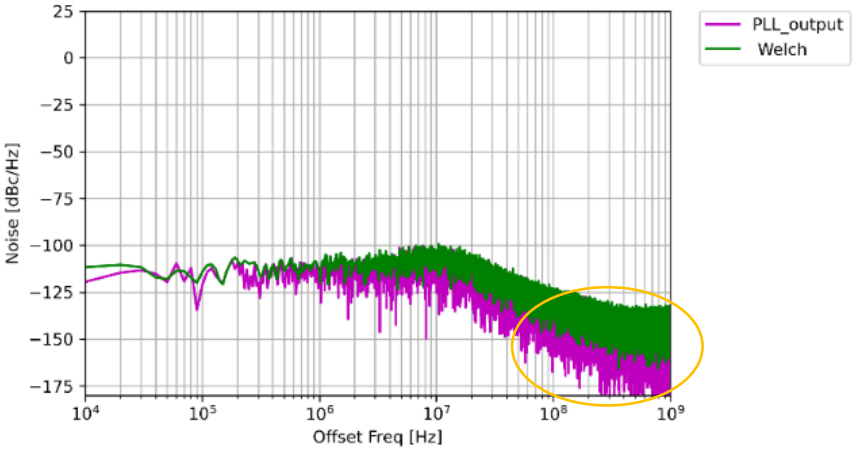

Figure 3.34: Phase noise of the CDR with the DCO phase jitter (100fs) applied to the system. The jitter contribution is 293.26fs.

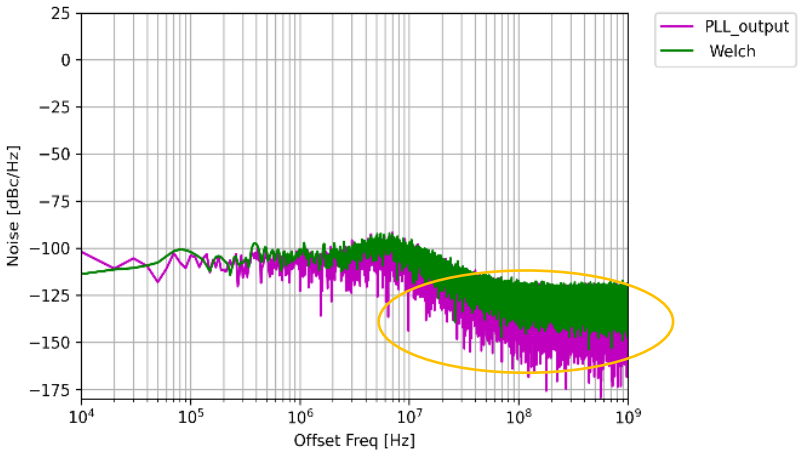

Figure 3.35: Phase noise of the CDR with the DCO phase jitter (500fs) applied to the system. The jitter contribution is 529.02fs.

The second source of noise considered in this case is the period jitter, for which a larger contribution is expected to cause an increase in the $1/f^2$ contribution. By comparing two cases (period jitter = 5fs) in Figure 3.36 and (period jitter = 50fs) in Figure 3.37 is it visible, also highlighted with the blue circle, that also in this case the contribution can be significant for larger sigma.

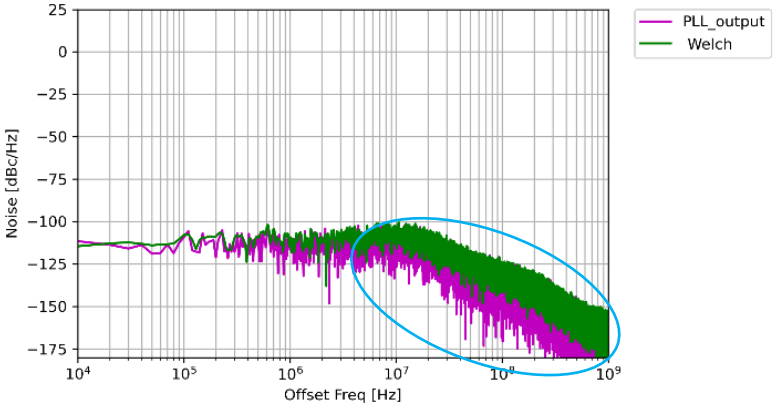

Figure 3.36: Period noise of the CDR with the DCO period jitter (5fs) applied to the system. The jitter contribution is 273.97fs.

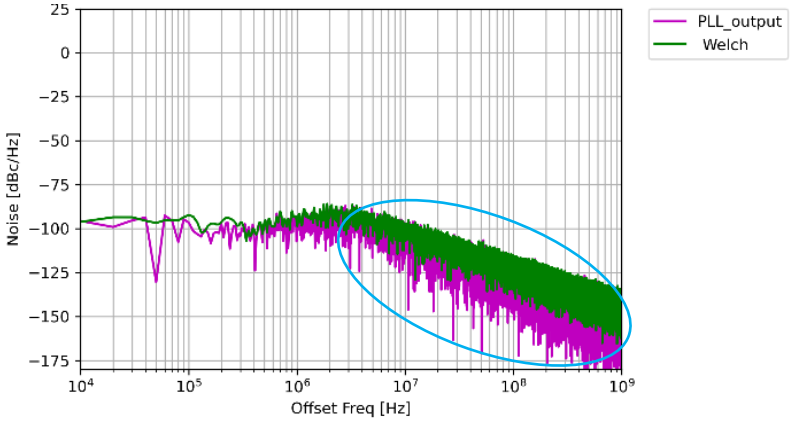

Figure 3.37: Period noise of the CDR with the DCO period jitter (50fs) applied to the system. The jitter contribution is: 889.05fs

The third noise contribution is the reference phase noise. In this case, the noise is mainly visible at lower frequencies. By comparing the two cases σ= 100fs and σ= 1ps in Figures 3.38 and 3.39, a larger reference noise corresponds also to a larger noise contribution in the highlighted area.

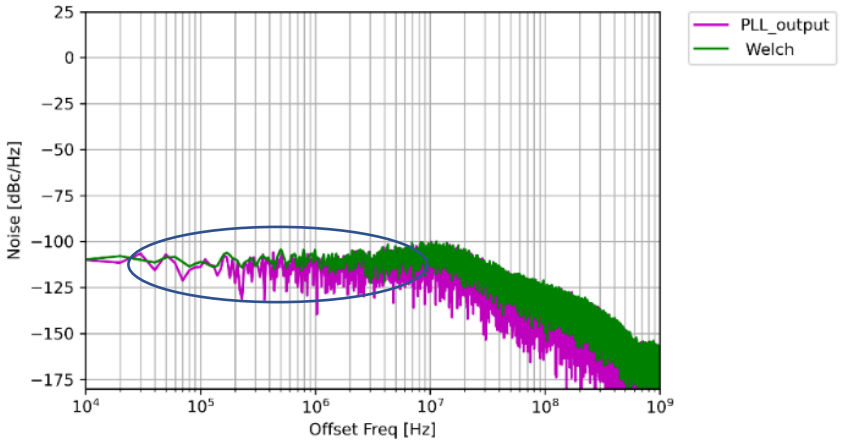

Figure 3.38: Reference noise on the CDR (σ= 100fs) applied to the system causes a jitter: 286.81fs.

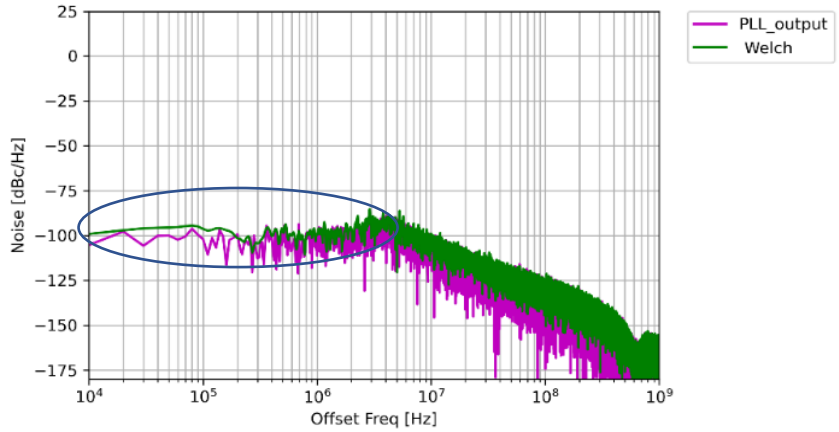

Figure 3.39: Reference noise on the CDR (σ = 1ps) applied to the system causes a jitter = 690.99fs.

3.12 Phase Transfer Function

As already described in Chapter 2, the study of the transfer function represents a useful approach to understanding how the noise propagates through the PLL. In particular, the PLL behaves as a low-pass filter and rejected the high frequencies jitter coming from the reference, while it works as a high-pass filter, attenuating the VCO jitter at low frequencies. As a result, the bandwidth is a useful parameter to estimate the intersection point between high-pass and low-pass behaviour. In addition, from the transfer function analysis, it is possible to evaluate the stability of the system by estimating for instance the damping factor and the phase margin.

By adding modulated delay which is shaped as a sinusoidal function with 3ps amplitude and by changing the oscillation of the delay it is possible to characterize the transfer function of the CDR, as shown in Figure 3.40. Furthermore, a random delay contribution of σ=250fs is also introduced.

Note that the phase transfer function is calculated as:

$$|H(s)| = \frac{\theta_{out}}{\theta_{in}} \quad (2.12)$$

As a result of the operation of the CDR, the input data stream is expected to be related to a clock of 2.5Gbps, while the output is targeted to be 12.5 GHz. Therefore, a gain of 5 is obtained in the transfer function, which corresponds to 14dB.

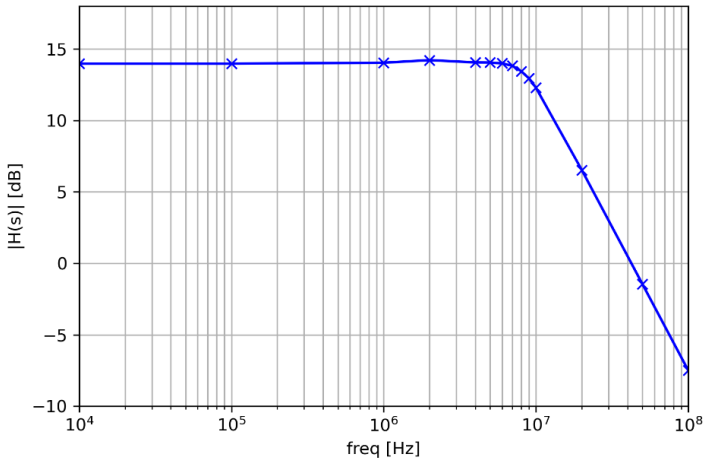

Figure 3.40: Phase transfer function of the CDR obtained interpolating simulation data.

3.13 Conclusion

In this chapter, the architecture and the characterization of a PLL-based CDR were shown. The proposed architecture finds application in a High-Speed serial interface, that takes as input the data stream with a 2.5 Gbps data rate and synthetizes an output frequency targeted to be 12.5 GHz.

A general description of the building blocks is given, as well as a representation of its dynamics in the time domain, with the definition of the time to lock, and amplitude of the oscillation as well as the locking range. Furthermore, a representation of the noise behaviour is given, considering the contribution of phase jitter and period jitter generated in the DCO as well as the noise contributed by the reference data input. Finally, the transfer function of the system is extracted.

Chapter 4 - CDR Operation with different Downsampler logic

4.1 Introduction

The trend of developing communication systems for high-speed transmission brings some limitations in the design of digital components. For this reason, alternative and effective solutions must be considered to maintain high timing accuracy and low noise in the design of PLL-based CDRs.

A downsampler plays an important role in the operation of the CDR since together with the PD defines the gain of the PD (K_{PD}). By changing the K_{PD}, it is indeed possible to change the dynamic of the CDR. The downsampler is used to reduce the operation speed of some blocks. When used at a high input data rate, such as 2.5 Gbps, instead of using the effective baud rate, the PD decision considered is at a lower rate. Operating at the data-rate speed may become problematic, and high energy demanding, such that the effort required to achieve such performance might be not necessary. Operating at a lower rate can be a better option in terms of power consumption and circuit speed requirements at the cost of a higher latency which might come at the cost of a less precise feedback-controlled system. Indeed, depending on the downsampler logic adopted, the PLL will have a different convergence range.

The analysis in Chapter 3, utilized a simple version of the downsampler. The operation of this block is based on collecting the PD decisions and processing the information in frames of eight samples. The dynamic of the CDR is dynamically adjusted according to the downsampler outcomes. The 1/8 samples logic represents the simplest implementation, but maybe not the best one in terms of noise since 7/8 samples are ignored. A reduction in speed results also in limited power consumption and limited problems in the circuital implementation deriving from the parasitic components and speed limitation.

A comparative analysis of three possible implementations of the downsampler will be shown below. The three solutions are compared in terms of PD gain and jitter when used in the CDR.

4.2 Downsampler with '1/8 Samples' Method

The working principle of this block was already described in Chapter 3. This downsampler is the simplest version of the three proposed in this chapter, which operates only one of the eight samples generated by the PD, discarding the successive seven samples and reducing the operation frequency, such that the PD works at the speed of the Input data (2.5 GHz) while the output data is generated at 312.5 MHz. Furthermore, this block aims to transform the output of the PD and convert it into three possible values (-1, 0, 1). This downsampler discards 7/8 samples, which can be different from the first sample. For example, considering a frame of eight PD decisions, like this set (-1, 1, 1, 1, 1, 1, 1, 1), the first sample is opposite to the successive seven. Considering this set of PD decisions, according to the 1/8 samples downsampler logic, only the first value would be taken into account. As a result, when used the first sample, the CDR will apply a correction to the DCO that is not optimal.

An example of operation is given in Figure 4.1

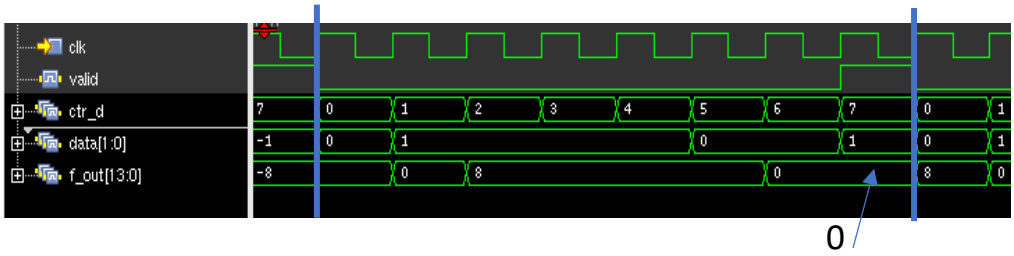

Figure 4.1: Example of operation of the downsampler with 1/8 samples logic. Picture taken from simulation with SimVision.

4.3 Downsampler with 'Most Frequent Decision' Method

The Most Frequent Decision (MFD) method provides the decision that is more recurrent in the observation frame, such that better control is used to control the DCO. For each window of eight phase detector decisions, this downsampler implementation will output the most frequently observed decision for downstream processing by the loop filter. This approach has the advantage of providing better control when used in the PLL. At the same time, it may come with the disadvantage of this logic comes at the cost of more complex logic. The most frequent decision logic is implemented as the sum/subtraction of eight samples, depending on the PD decision. However, if the result of the sum is larger than ±1 the output is saturated to ±1.

As well as the previous case, it reduces the operation frequency, such that the PD works at the speed of the Input data (2.5 GHz) while the Output data is generated at 312.5 MHz. An example of an operation is given in Figure 4.2. In particular, 'data[3:0]' gives an example of PD decisions collected. The samples under study are the values of 'data[3:0]' when the downsampler counter, 'ctr_d', is in the range (0-7). The result is then shown when the 'valid' bit goes 1. In the example hereby reported, eight samples are collected, and their values are iteratively summed. The result of the sum is then converted to the maximum acceptable value. In practice, -4 is then converted to -1. To maintain a coherent data structure among the three topologies the last 3 bits of the output are considered as fractional values. However, the MFD downsampler generates only integer values, therefore, -1 will corresponds to -8 as the output is shown in decimal scale.

It is expected that this downsampler will provide better control when used inside the PLL since the result generated is a combination of the eight decisions. This approach will indeed avoid a mistaken decision, as the one reported in the previous case. However, as a pitfall, according to the approach used to implement this logic, considering a frame of eight PD decisions, like this set (0, 0, 1, 0, -1, 0, 0, 1), the output generated will be -1, as a result of the sum of the eight values. However, the most frequent value is 0.

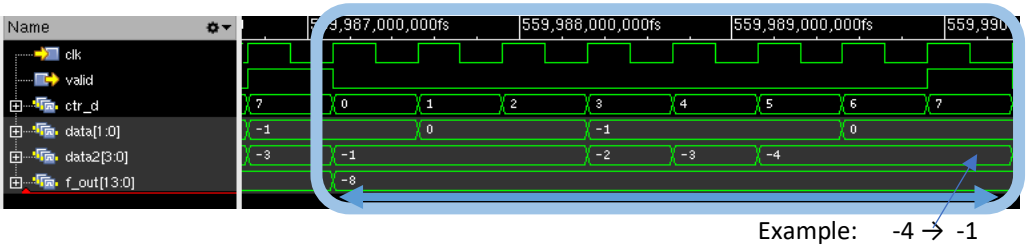

Figure 4.2: Example of operation of the downsampler with MFD logic. Picture taken from simulation with SimVision.

4.4 Downsampler with 'Average' Method

Another possible implementation of the downsampler is to calculate the average (AV) of the eight samples collected in an observed time frame. The average is calculated with the bitwise operator, [31], which consists in computing the sum of the eight samples and shifting the output of 3 bits to the right to accomplish the division over 8. As a result of the average calculation, unlike the other two cases, fractional results are allowed. An example of operation is given in Figure 4.3

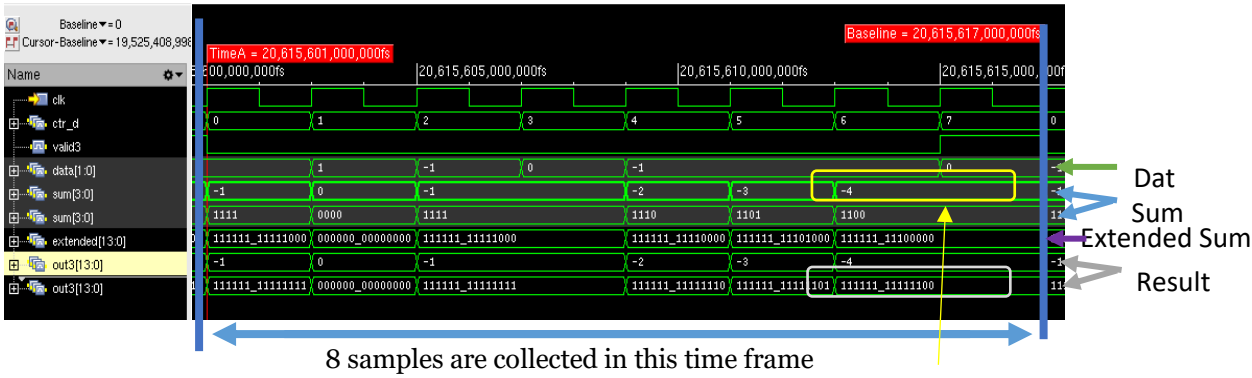

Figure 4.3: Example of operation of the downsampler with average logic. Picture taken from simulation with SimVision.

The data generated is extended to a 14-bit fixed point format to shape the data as shown below:

zzzzzzz_zyyy_xxx

[Sign Extension] [Data] [Fractional]

The data shown in Figure provides -4 as a result of the sum of the eight samples. Therefore, the average is expected to be -4/8=-0.5 which considering an output string of 14 bit with 3 fractional numbers gives: 11111111111_100.

4.5 Downsampler Characterization

The downsampler used in combination with the PD will have a specific gain (K_{PD}) and transfer function, which can assume different values depending on the downsampler logic implemented. As already mentioned, the BBPD is a non-linear detector which provides information as Early/Late in case of transition, which can be interpreted as a time-to-digital quantizer. The Phase detector error function is obtained by observing the output of the downsampler by applying random noise around every rising edge. A linearized transfer function can be obtained considering small input delay variation. The phase detection curve (S-Curve) of the PD is obtained as the mean of the output of the PD, processed according to the operation of the downsampler. Figure 4.4 shows the testbench used to characterize the operation of the three downsamplers. The input data generated for the three downsamplers are the same, as well as the PD block.

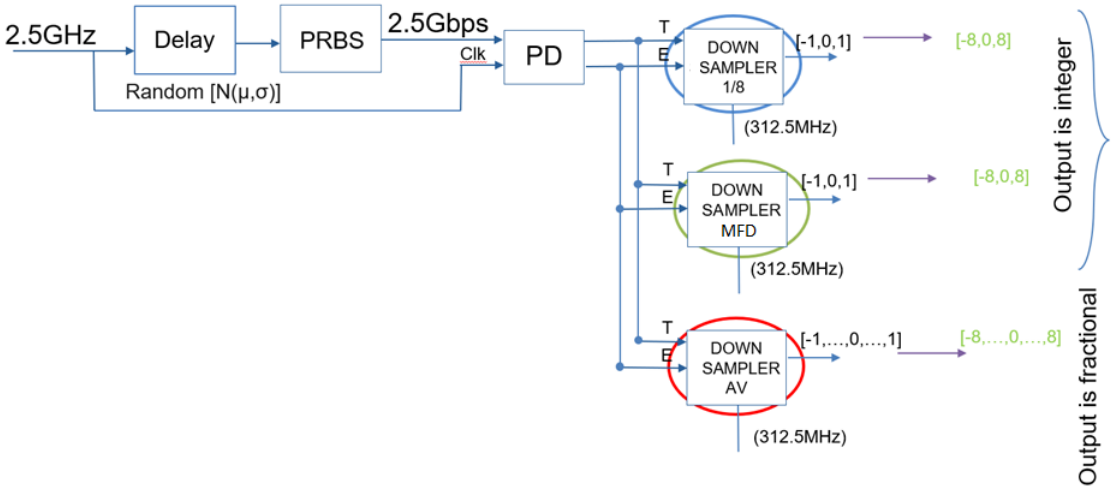

Figure 4.4: Testbench used to characterize the downsamplers.

Figure.4.5 illustrates the S-Curve of the three implemented topologies of the downsampler. By measuring the output and counting the number of commutations at zero crossings of the output the gain can be estimated according to Equation 2.4. By calculating the gain in zero:

- $K_{PD\,MFD} = 3.1 \cdot K_{PD\,1/8}$
- $K_{PD\,1/8} \approx K_{PD\,AV}$

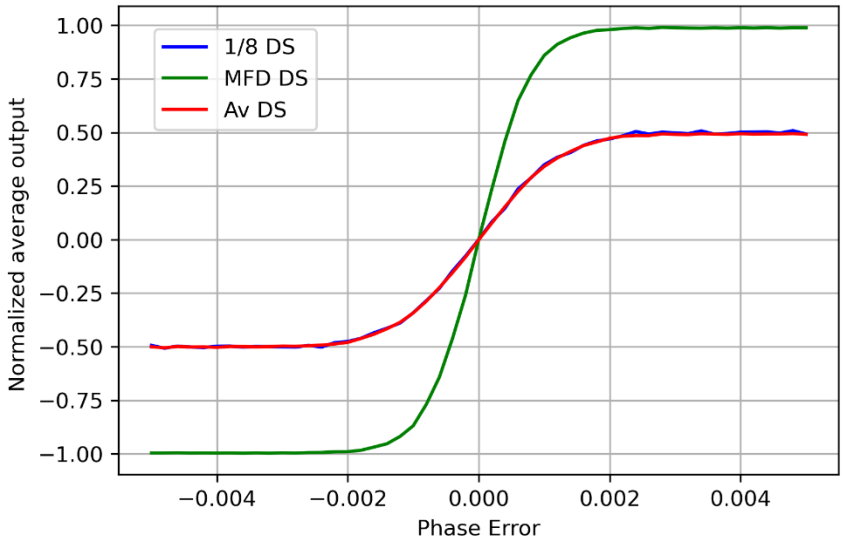

Figure 4.5: Simulated S-Curve of the PD and downsampler.

Two of the downsampler topologies, 1/8 samples and average, seem to have a roughly equivalent behaviour, while the Most Frequent Decision (MFD) seems to have a higher gain as well as a scaled output. Furthermore, considering the two equivalent solutions the average seems to have a much flat behaviour interpreted as less noisy PD, which is preferable.

4.6 CDR Architectures Comparison

A fair comparison between the three CDR topologies will be presented below. The comparison is done considering the phase transfer function, taking into account the same setup, testbench and filter coefficients for the three systems, as shown in Figure 4.6. A different implementation of the downsampler provides a variation in the PD (K_{PD}) and as a consequence also a different transfer function. According to Equations 1.9, and 1.10, this variation in the gain reflects in the properties of the CDR, such as the natural frequency (ω_n), the damping factor (ζ) and the jitter.

The fitting of the transfer functions is obtained considering a 1st-order loop filter modelled using the same two known parameters (ω_n=35 MHz) and (ζ=4) for the three curves, but different K_{PD}, according to Equations 2.16 and 2.17. The 1/8 samples downsampler CDR which is represented in blue, is used as a reference system, while the two other curves are obtained fitting a proportion in the PD gain (K_{PD}). As a result, the CDR with MFD downsampler presents a $K_{PD\_MFD} \approx 2.75 \cdot K_{PD\_1/8}$, while the CDR with Average downsampler has a $K_{PD\_AV} \approx 0.55 \cdot K_{PD\_1/8}$.

Note that the PD is a non-linear system, and the transfer function is a linearized representation of the system. Some discrepancies between the theoretical model and the simulated system can be visible in Figure 4.6. It is possible to have a good match at low and high frequencies, but a larger misalignment is visible around the cutoff frequency. The filter coefficients in this case are:

- K_1 = 14'b000000_11111101;
- K_2 = 14'b000000_00000011;

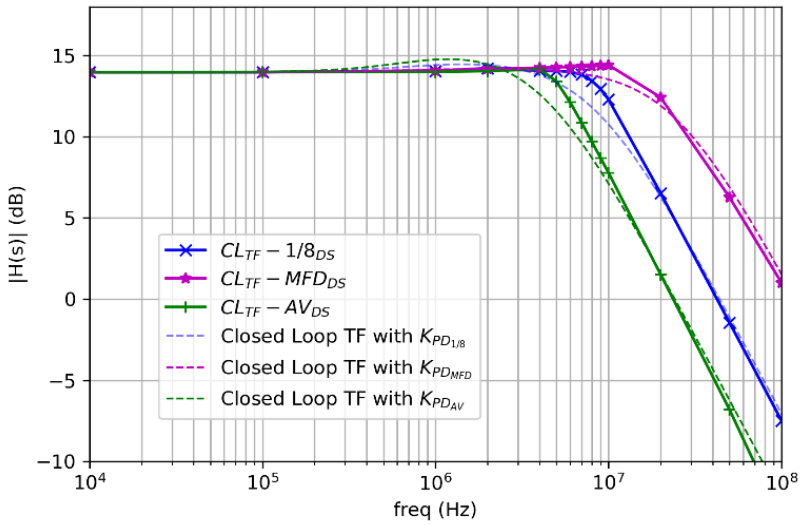

Figure 4.6: Calculated phase transfer function theoretical and interpolated with simulation data. In the theoretical model it is assumed (ω_n=35 MHz) and (ζ=4).

As a next step, the dynamic of the three topologies, shown in Figure 4.7, 4.8, and 4.9, are compared, in which the time to lock and the oscillation amplitude is considered.

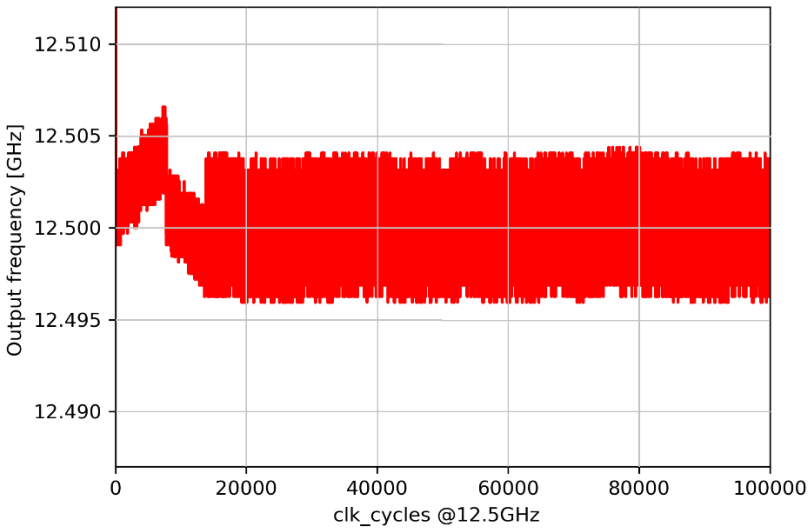

Figure 4.7: Dynamic of the CDR with 1/8 samples downsampler. (Output frequency vs number of clock cycles)

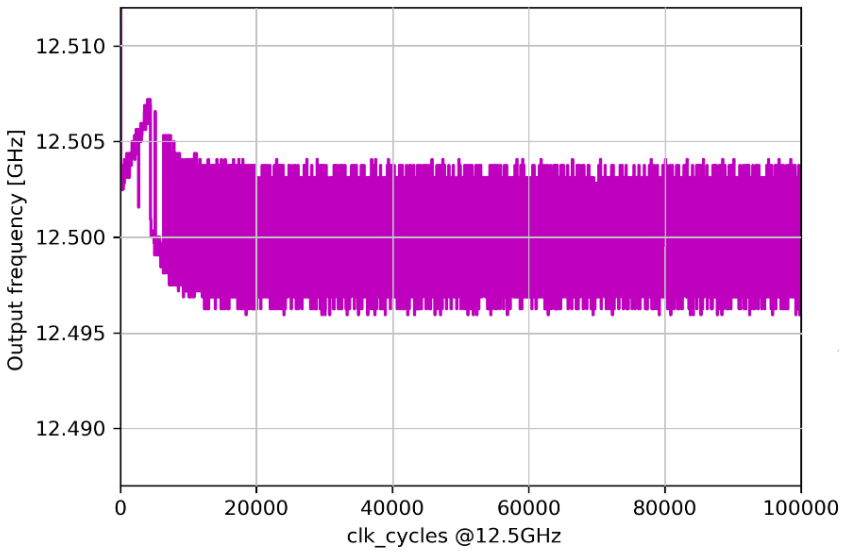

Figure 4.8: Dynamic of the CDR with MFD downsampler. (Output frequency vs number of clock cycles)

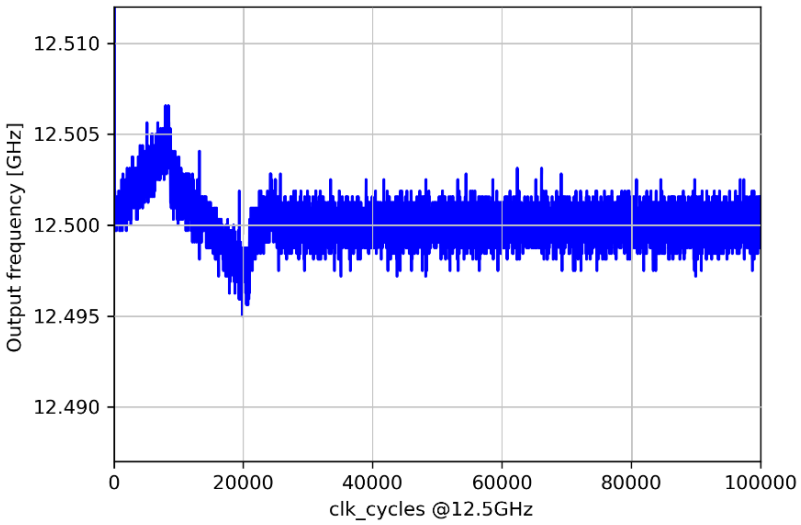

Figure 4.9: Dynamic of the CDR with AV downsampler. (Output frequency vs number of clock cycles)

The CDR with MFD downsampler seems to be the fastest to lock, while the CDR with AV downsampler is the slowest but the one with smaller output frequency oscillation once the CDR is locked. On the other hand, the CDR with 1/8 samples downsampler is the simplest solution.

Every design choice comes with advantages and drawbacks that should be evaluated case by case depending on the system constraints and the application. In the end, the three models seem to be a valid alternative since, it is possible to compensate for the variation of the PD gain by selecting different K_1, and K_2 coefficients, making in principle the behaviour of the three solutions equivalent, as shown in Figure 4.10. For the purpose of this thesis, the jitter contribution is the most important parameter, therefore, to compare the three CDR topologies, the filter coefficients are modified for the three CDR the same transfer function is obtained. In this case, the jitter is calculated for each CDR architecture, trying to obtain systems with the same ω_n and ζ.

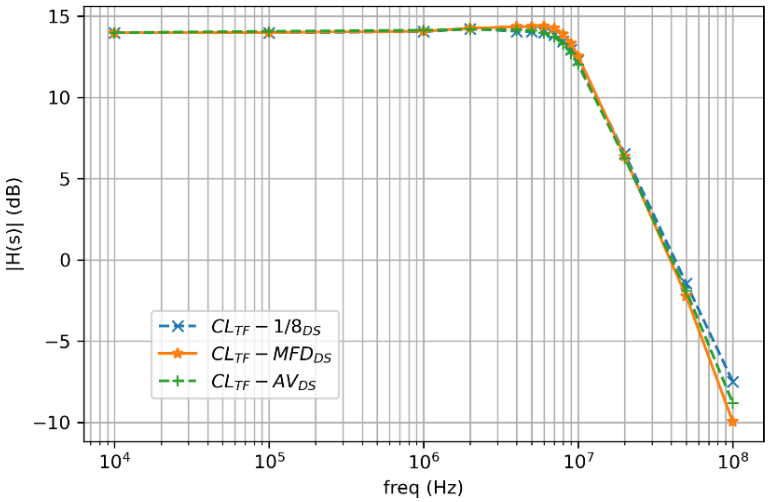

Figure 4.10: Obtained phase transfer function considering different K_1 and K_2 coefficients in the filter for each CDR topology.

As a result, to obtain the same transfer function for the three CRDs the following filter coefficients have been chosen and the jitter has been calculated accordingly:

- (K_1 = 14'b000000_11111101, K_2 = 14'b000000_00000011)
  Jitter on CDR1 (1/8): 183.75fs.
- (K_1 = 14'b000000_01111101, K_2 = 14'b000000_00000011)
  Jitter on CDR2 (MFD): 208.90fs, which corresponds to +14%, w.r.t. CDR1.
- (K_1 = 14'b000001_10111101, K_2 = = 14'b000000_00000011)
  Jitter on CDR3 (AV): 142.52fs, which corresponds to -22%, w.r.t. CDR1.

DOWNSAMPLER	**1/8**	**MFD**	**AV**
Complexity	Simplest	Moderate	Moderate
Normalized K_{PD}	1	2.75	0.55
Considering the same filter coefficients (K1, K2)			
CDR Bandwidth	10 MHz	20 MHz ≈ +100%	6 MHz ≈ -40%
Time to lock	≈1.36 µs	≈800 ns ≈ -41%	≈2 µs ≈ +47%
Jitter	183.75fs	331.231fs	146.29fs
Considering the same transfer function			
CDR Bandwidth	10 MHz	10 MHz	10 MHz
Time to lock	≈1.36 µs	≈1.2 µs ≈ -11%	≈1.35 µs ≈ -0.7%
Jitter	183.75fs	208.90fs ≈ +14%	142.52fs ≈ -22%

Table 4.1: Comparison of the three CDR topologies.

4.7 Conclusion

A fair comparison between the three topologies is proposed, by comparing the phase transfer function, the jitter and the locking time to evaluate the performance of the three systems. Depending on the dynamic of the CDR, a different time to lock can be observed, as well as a different bandwidth and a noise contribution and jitter among the three systems.

The comparison among the three topologies is initially done taking into account the same setup, testbench and filter coefficients. As a result, if the same filter coefficients are used, the CDR using the MFD downsampler has the largest bandwidth and therefore the fastest time to lock. While the AV is the slowest to lock, once locked has the smallest tuning range around the targeted frequency.

A second comparison is done considering the same setup, and testbench but different filter coefficients to provide the same transfer function. As a result, the AV CDR is the less noisy (-22%), while the MFD is the noisiest (+14%)

All three models seem to be a valid alternative and their behaviour can be customised by changing the filter parameters such that their behaviour in terms of transfer function is similar.

As a result, an optimal solution is not given, since every design choice comes with advantages and drawbacks that should be evaluated case by case depending on the system constraints and the application, if a lower noise is required, an AV downsampler is preferred, while, if a CDR with larger bandwidth is desired, the MFD downsampler is desirable, and if a simpler implementation is targeted the 1/8 samples downsampler is the best choice.

Conclusions

An All-digital PLL-based architecture for CDR was presented. The proposed topology can be used to support the development of ASICs compatible with the clock distribution requirements of High Luminosity Large Hadron Collider (HL-LHC) detectors. The proposed architecture represents a valid alternative system to generate a clock signal of 12.5 GHz starting from an input signal with a data rate of 2.5 Gbps.

Chapter 1 provided basic information about the operation of the CDR and presented the last trends in the state-of-the-art of PLLs. In Chapter 2 a PLL architecture was studied considering a linear time-invariant modelling, through the phase transfer function analysis to describe the behaviour of the CDR and the noise propagation across the system. It was found that the PLL behaves as a low-pass filter and rejected the high frequencies jitter coming from the reference, while it works as a high-pass filter, attenuating the VCO jitter at low frequencies. As a result, the bandwidth is a useful parameter to estimate the intersection point between high-pass and low-pass behaviour.

In Chapter 3 a block diagram representation of the proposed PLL-based CDR is given, analysing the operation and the properties of each block. All in all, the key aspects for designing and understanding a PLL-based CDR are given, and the advantages of a digital architecture are summarized. The proposed study gives a comprehensive description of the dynamic of the CDR, of which are the main noise sources, such as the reference data stream and the DCO, as well as how their contribution impacts and propagates across the feedback system. The CDR presented in this chapter forms the basis of the following analysis, which focuses on the implementation of the downsampler.

Finally, chapter 4 compares three topologies where the downsampler is the block that operates differently. The comparison is done in terms of PD gain, jitter and dynamic of the CDR. Depending on the application and its requirements the best choice can vary. The three models seem to be a valid alternative and their behaviour can be customised by changing the filter parameters such that their behaviour is similar. The simplest solution is the 1/8 samples downsampler, the less noisy is the one with an Average downsampler, while for larger bandwidth the best solution is the one with an MFD downsampler.

Future Work

High-level behavioural analysis of the proposed system has been discussed, however, the synthesis and the implementation at the transistor level of the studied CDR would be a starting point to validate the ideas shown and support the drawn conclusions. In addition, further problems such as the delay variation of the input signal, and the synchronization of the subblocks should be considered such that a more realistic system will be studied.

The presented results seem to be promising, however, due to a limited time the implementation of the CDR on silicon was not possible but remains an interesting topic for future works to have a more comprehensive understanding of this topic.

Furthermore, from the point of view of the hardware implementation, most likely the LC-DCO is the most challenging block, which needs to be optimized for low noise and to reduce the effects of parasitic capacitances, as well as limit the occupied area and reduce the power consumption, therefore a topology study is strongly recommended.

Appendix A

Second Order Filter Modelling

Depending on the requirements of the CDR, it is possible to select a filter with higher selectivity to further reduce high-frequency contribution. In particular, Z(s) can be a 1st filter as shown in Chapter 2, but if needed can be modelled as a higher-order filter.

This study also aims to compare the advantages and drawbacks of a higher-order filter therefore, a 2nd order filter is considered.

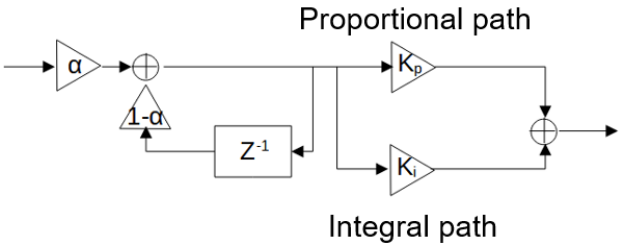

Figure A.1: Block diagram of a 2nd-order filter.

In particular, the 2nd order filter corresponds to a 3rd system. The 2nd order filter, shown in Figure A.1 can be modelled with the following transfer function:

$$Z_2(s) = \frac{s\cdot\tau_2 + 1}{s\cdot\tau_1(s\cdot\tau_3 + 1)} \quad (A.1)$$

Considering a 2nd order filter $Z_2(s)$ the closed loop transfer function can be written as:

$$TF_{CL}(s) = \frac{TF_{OL}(s)}{1+TF_{OL}(s)\cdot H(s)} = \frac{\frac{K_{PD}\cdot K_{DCO}}{s}\cdot Z_2(s)}{1+\frac{K_{PD}\cdot K_{DCO}}{s}\cdot Z_2(s)} = \frac{\frac{K_{PD}\cdot K_{DCO}}{s}\cdot\frac{s\cdot\tau_2+1}{s\cdot\tau_1(s\cdot\tau_3+1)}}{1+\frac{K_{PD}\cdot K_{DCO}}{s}\cdot\frac{s\cdot\tau_2+1}{s\cdot\tau_1(s\cdot\tau_3+1)}} = \quad (A.2)$$

$$TF_{CL}(s) = \frac{K_{PD}\cdot K_{DCO}\cdot(s\cdot\tau_2+1)}{s^2\cdot\tau_1(s\cdot\tau_3+1) + K_{PD}\cdot K_{DCO}\cdot(s\cdot\tau_2+1)} \quad (A.3)$$

By displaying the transfer functions described in equations 2.4- 2.6, it is possible to study the behaviour of the third order system. To compare the behaviour of the 1st-order filter and the 2nd-order one, the transfer functions are represented with the same colour but different shades. In particular, in Figure A.2 the curves blurry refer to a system with a 1st order filter.

As already mentioned, the blue line represents the transfer function of the low pass filter, for which at a higher frequency a higher selectivity is obtained. The purple line is the open loop transfer function, the green line is the DCO transfer function and the yellow one is the closed loop transfer function.

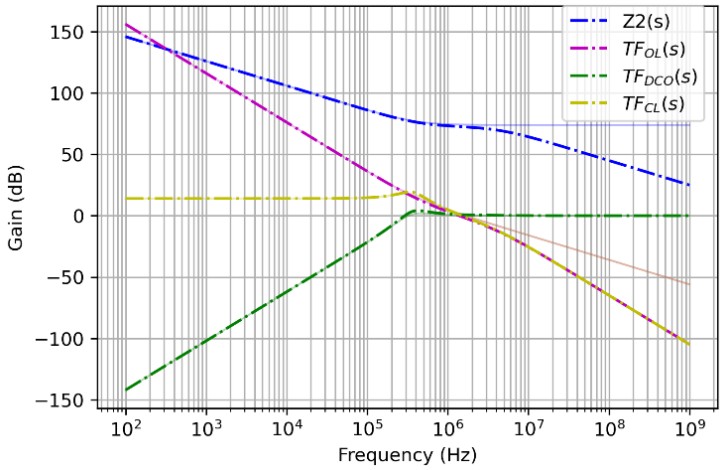

Figure A.2: Transfer function of a CDR with a 2nd order filter.

Considering the same approach proposed for the 1st filter, the closed loop transfer function (Equation A.3) can be written as:

$$TF_{CL}(s) = \frac{K\cdot(s\cdot\tau_2+1)}{s^3\cdot\frac{\tau_2^3}{b}+s^2\cdot\tau_2^2+K_1\cdot s\cdot\tau_2+K_1} \quad (A.4)$$

where it is possible to model his behaviour using 3 parameters:

$$K = \frac{K_{PD}\cdot K_{DCO}\cdot\tau_2}{\tau_1} \quad (A.5)$$

$$b = \frac{\tau_2}{\tau_1} \quad (A.6)$$

$$K_1 = K\cdot\tau_2 \quad (A.7)$$

As in the previous case, K behaves similarly to ω_n, therefore considering b=9, K_1 = 3, it is possible to choose a large K to minimize the jitter, for instance, K > 100 MHz is preferable, as shown in Figure A.3.

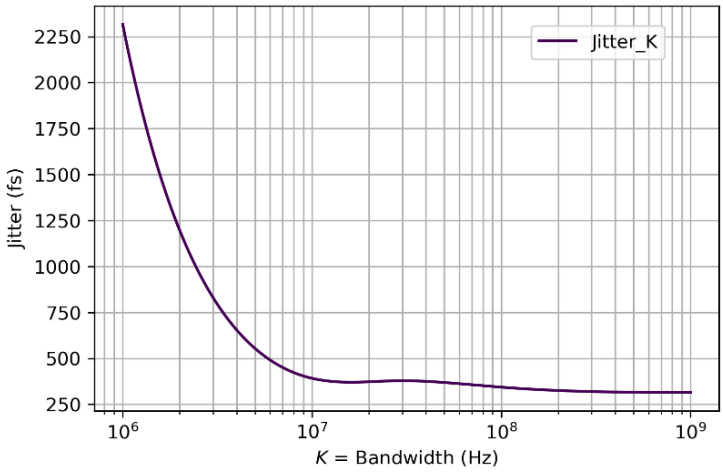

Figure A.3: Parametric analysis of K, [b=9, K_1 = 3].

By considering a fixed b=9 and calculating a parametric sweep of the K_1 it is possible to estimate the optimal K to minimize the jitter. To minimize the jitter using a 2nd order filter, large b is preferred and $K_1 \approx$ (1-10), as shown in Figures A.4 and A.5.

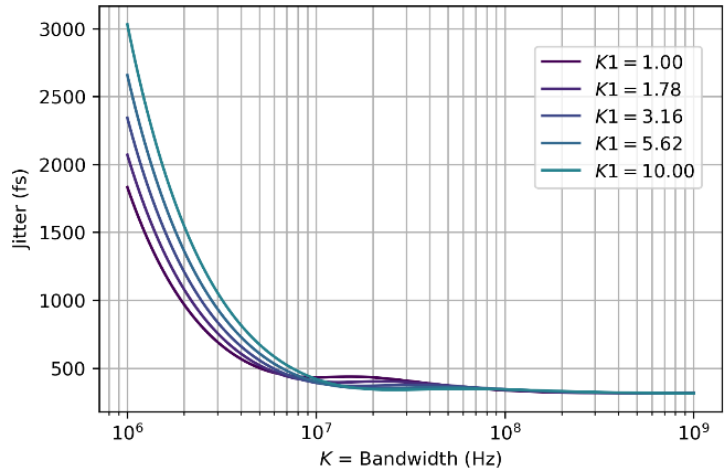

Figure A.4: Parametric analysis of K_1 and K, with b=9.

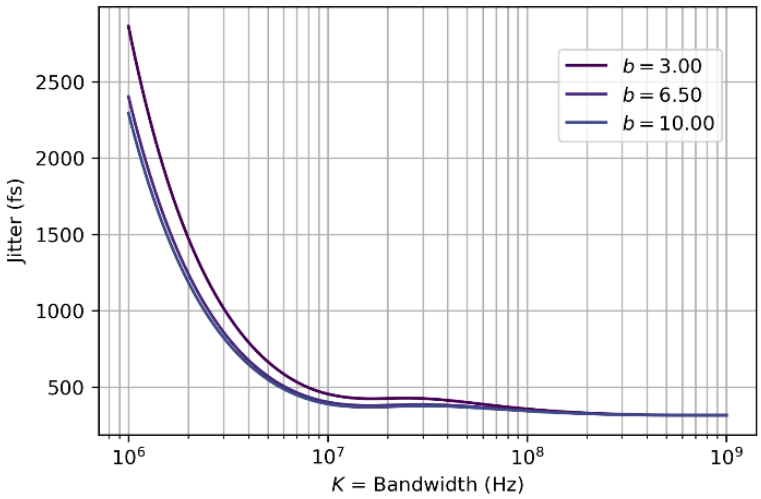

Figure A.5: Parametric analysis of K and b, and $K_1 = 3$.

Once defined the transfer function of the system it is necessary to model the phase noise to study how it propagates through the system and how it is sufficiently attenuated.

Following the analysis of Chapter 2, the total phase noise which is constituted by the DCO noise and phase noise coming from the reference data stream are considered. The resulting phase noise, obtained as a linear combination of the two noise sources is represented by the blue line in Figure A.6. In particular, the convention chosen assumes that the dashed line is the result of a 1st filter (2nd order feedback system), while the continuous line is referred to a 2nd order filter (3rd order system).

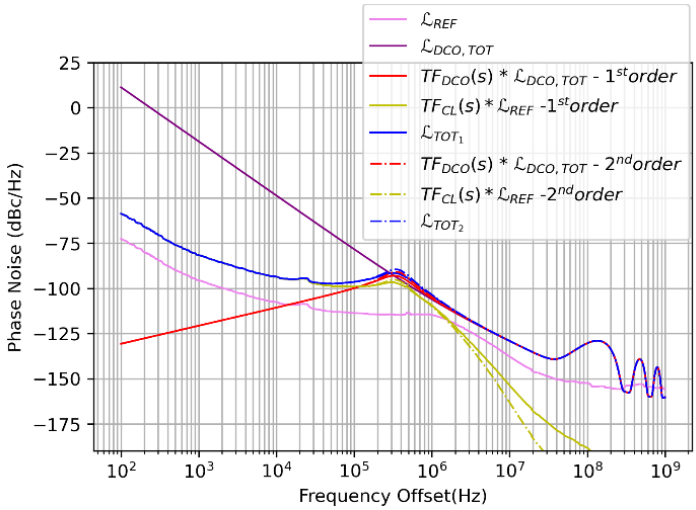

Figure A.6: Comparison of transfer function analysis between a 1st-order filter and a 2nd-order filter, considering a small reference noise contribution.

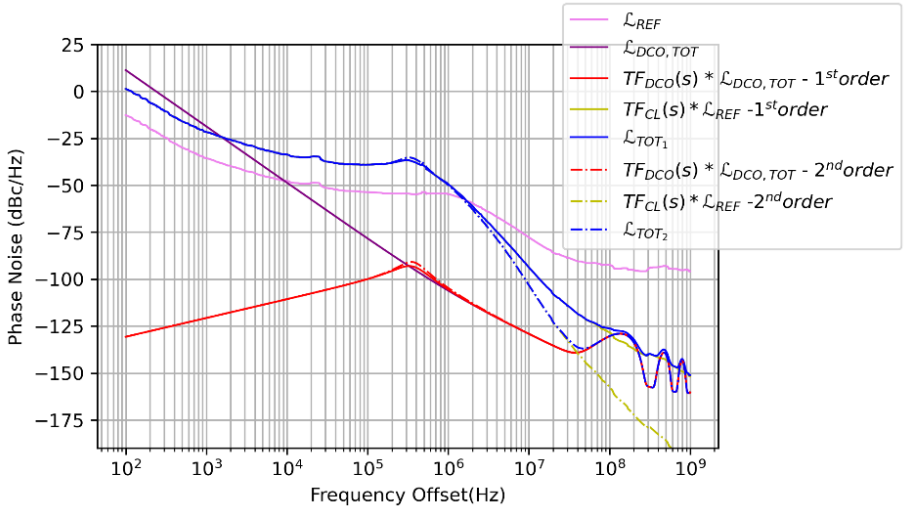

Figure A.7: Comparison of transfer function analysis between a 1st-order filter and a 2nd-order filter, considering a large reference noise contribution.

It is possible to conclude that using a more selective filter can be beneficial in case the reference noise is the dominating contribution. As shown in Figure A.7, if the reference noise is significantly larger, its effects are mainly visible at higher frequencies, due to the intrinsic behaviour presented in Chapter 2. As a result, since the selectivity of the filter reflects also the selectivity of the closed loop transfer function, the contribution at higher frequency can be further suppressed. This is highlighted by comparing Figures A6 and A7, in which the pink line represents the reference noise contribution. In the second case, the reference noise is intentionally shifted high to make more evident the filter properties.

In addition, it is also worth mentioning that a higher-order filter provides a more selective behaviour in the feedback system, which is of course an advantageous property. On the other hand, a more complex filter comes with other technical issues.

Therefore, it is important to conclude that always a trade-off between benefits and higher complexity, and occupied area, must be considered before a decision is made in the design of a CDR.

Appendix B

RMS Phase Jitter Calculation

In high-speed transceivers, phase noise is targeted to be as low as possible. Phase noise is indeed an undesired variation in the phase of a signal caused by small fluctuations in the signal phase and is expressed in units of dBc/Hz. This effect in the time domain translates into small fluctuations in the phase of the signal around the ideal clock signal. Phase noise and Jitter are strictly related since a random variation in the phase of a signal can also be analysed as a fluctuation of the edges of a clock signal with respect to the nominal expected behaviour and is generally expressed in ps.

When observed in the frequency domain, the phase noise is noticed as a spread of the harmonics and in general in a higher noise level and a worsening in the system performance which translates into a high Bit Error Rate and corruption in the transmitted data.

Another important parameter to consider studying the noise contribution inside a CDR is the RMS phase jitter, [26]. More in specific, the phase noise is used to calculate the jitter in the frequency domain and to estimate the RMS Phase Jitter value which gives a measure of quality and spectral purity.

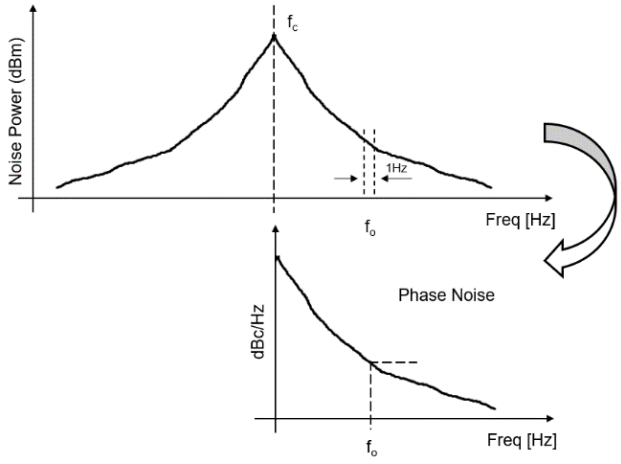

Figure B.1: Qualitative phase noise representation.

In the ideal case, the harmonic contribution of a signal oscillating at a nominal frequency f_c should be represented as a pulse in the defined frequency, however, the non-idealities, in the system can cause small changes in the oscillation frequency that swings around the carrier frequency generating a sideband such that the frequency spectrum oscillates in $f_c \pm f_o$, resulting in the undesired effect called phase noise, which is shown in the Figure B.1. When the noise power is measured in 1Hz, band at an offset frequency f_o and referred to a carrier frequency f_c the resulting graph is measured in dBc/Hz, resulting in what was previously modelled by Leeson's equation ($\mathcal{L}(f)$), as shown at the bottom. From Leeson's equation, it is possible to calculate the RMS phase noise $\left(\sqrt{2 \cdot \int_a^b \mathcal{L}(f)\, df}\right)$ and the jitter as shown in Equation B.1

$$\sigma_t = \frac{\sqrt{2 \cdot \int_a^b \mathcal{L}(f)\, df}}{2 \cdot \pi \cdot f_{osc}} \quad \text{(B.1)}$$

Where a and b represent the integration range in frequency and f_{osc} is the oscillation frequency of the oscillator.

Appendix C

Study case: Constant input →3 bit input (Xi = 1)

To clarify the operation of a MASH ΣΔM, a study case of a 2 stages MASH, shown in Figures C.1, C.2 and C.3 is presented in this section. In this case, to maintain the description as simpler as possible it is assumed a constant 3-bit input (Xi).

Note that 3 input bits for a 2 stages MASH is not a good choice since the output generated is also 3bit, so this assumption is used only to have a simple case to study. Also, in the real cases, the input should change according to the fractional part of the filter output, therefore it is expected to be a sequence of 8bits.

The MASH is in this case constituted by 2 cascaded 1st order ΣΔM. The output is then processed by the NCL.

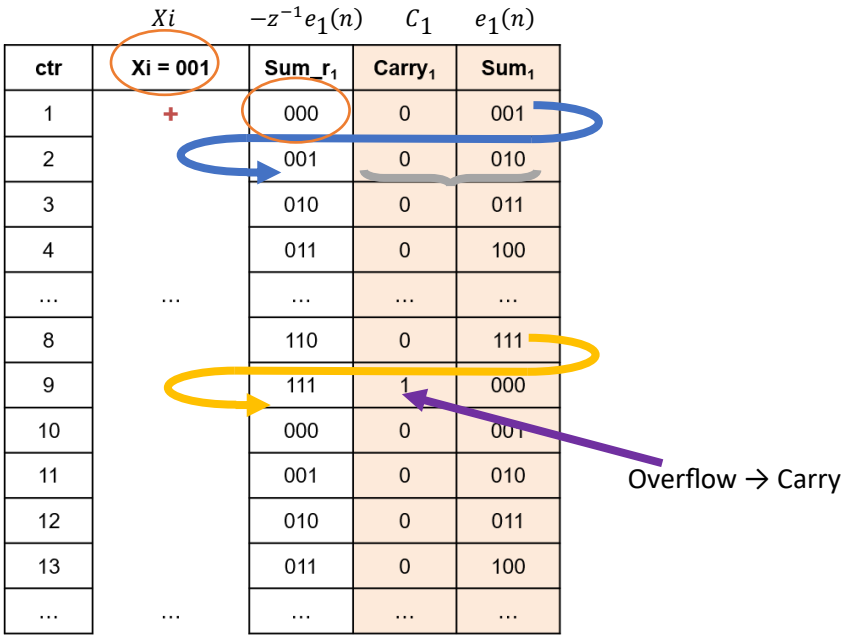

Figure C.1: Example operation of a first stage ΣΔM.

	1st stage			2nd stage			
ctr	Xi = 001	Sum_r_1	$Carry_1$	$Sum_1 = Xi_2$	Sum_r_2	$Carry_2$	Sum_2
1		000	0	001	000	0	001
2		001	0	010	001	0	011
3		010	0	011	011	0	110
4		011	0	100	110	1	010
5		100	0	101	010	0	111
6		101	0	110	111	1	101
7		110	0	111	101	1	100
8		111	1	000	100	0	100
9		000	0	001	100	0	101
10		001	0	010	101	0	111
11		010	0	011	111	1	010
12		011	0	100	010	0	110
13		100	0	101	110	1	011
14		101	0	110	011	1	001

Figure C.2: Example operation of the first stage and the second stage ΣΔM.

ctr	$Carry_1$	$Carry_2$	$Carry_2\_reg$	ΔN
1	0	0	x	x
2	0	0	0	0
3	0	0	0	0
4	0	1	0	1
5	0	0	1	-1
6	0	1	0	1
7	0	1	1	0
8	1	0	1	0
9	0	0	0	0
10	0	0	0	0
11	0	1	0	1
12	0	0	1	-1
13	0	1	0	1
14	0	1	1	0

C_1 + C_2 - C_{2reg}

Figure C3: Example operation of the Noise Cancelling Logic.

References

[1] Banerjee, D. "*PLL Performance, Simulation, and Design - 5th Edition*", Dog Ear Publishing, 2017.

[2] "*Application note - PLL Application*" http://www.iitk.ac.in/eclub/summercamp/Courses/Special%20Topics/self331_05pll_appl.pdf

[3] Biereigel, S.; Kulis, S.; Moreira, P.; Kölpin, A.; Leroux, P.; Prinzie, J. "*Radiation-Tolerant All-Digital PLL/CDR with Varactorless LC DCO in 65 nm CMOS*" Electronics 2021, 10, 2741. https://doi.org/10.3390/electronics10222741

[4] https://www.home.cern/about/who-we-are/our-mission

[5] T. Raubenheimer on behalf of FCC collaboration & FCCIS DS team, "*FCC Accelerator Overview - FCC Week Workshop*" -June 5, 2023 https://indico.cern.ch/event/1202105/contributions/5423512/attachments/2659506/4610093/230604%20FCC%20Week.pdf

[6] Wanotayaroj, C and Mendes, E and Baron, S and Kulis, S," DART28-FPGA implementation study for future high-speed links", JINST vol.18, n. 01, 2023, doi : 10.1088/1748-0221/18/01/C01024, https://cds.cern.ch/record/2861855

[7] Staszewski, Rafal, et al. "*A Digitally Controlled Oscillator in a 90 Nm Digital CMOS Process for Mobile Phones*". IEEE Journal of Solid-state Circuits, vol. 40, no. 11, Institute of Electrical and Electronics Engineers, Oct. 2005, pp. 2203–11. https://doi.org/10.1109/jssc.2005.857359.

[8] Walker, Richard. (2003). "Designing Bang-Bang PLLs for Clock and Data Recovery in Serial Data Transmission Systems. https://www.researchgate.net/publication/276295515_Designing_Bang-Bang_PLLs_for_Clock_and_Data_Recovery_in_Serial_Data_Transmission_Systems

[9] Bueren, Georg Paul Emil von, 2011, "Clock and data recovery circuit and clock synthesizers for 40 Gb/s high-density serial I/O-links in 90- nm CMOS", https://www.research-collection.ethz.ch/bitstream/handle/20.500.11850/152964/eth-4700-02.pdf

[10] Biereigel, S. 2022, "*Radiation-Tolerant All-Digital Clock Generators for High Energy Physics", PhD thesis, Brandenburg University of Technology* https://cds.cern.ch/record/2834952/files/CERN-THESIS-2022-139.pdf?version=1

[11] Application Note: AN-815 "Understanding Jitter Units" – Application note. https://www.renesas.com/us/en/document/apn/815-understanding-jitter-units

[12] Application Note: MT-008 "Converting Oscillator Phase noise to Time Jitter", https://www.analog.com/media/en/training-seminars/tutorials/MT-008.pdf

[13] Application Note: AN-03: SONET OC-12 JITTER MEASUREMENT. Rev. C. MicrelSynergy. June 1998. https://ww1.microchip.com/downloads/en/Appnotes/AN03-TCG.pdf

[14] Da Dalt N. 2007, "Theory and Implementation of Digital Bang-Bang Frequency Synthesizers for High-Speed Serial Data Communications", Aachen, Techn. Hochsch., Diss., 2007. PhD thesis. Aachen, 2007, XVI, 174 S.: graph. Darst. https://d-nb.info/985885246/34

[15] J. Sonntag, J. Stonick, “A Digital Clock and Data Recovery Architecture for Multi-Gigabit/s Binary Links” IEEE JOURNAL OF SOLID-STATE CIRCUITS, VOL. 41, NO. 8, AUGUST 2006, https://www.researchgate.net/figure/Early-late-phase-sampling_fig3_2983206

[16] A. M. Hussein, “Low Power, Sm, all Area,All Digital Phase Locked Loop ”, Bachelor Thesis, Cairo University, 2014, https://scholar.cu.edu.eg/?q=hmostafa/files/gp_2014_2.pdf

[17] Avallone, Luca, et al. “A Comprehensive Phase Noise Analysis of Bang-Bang Digital PLLs.” IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, vol. 68, no. 7, July 2021, pp. 2775–86. https://doi.org/10.1109/tcsi.2021.3072344

[18] Verbeke, Marijn, et al. “Inverse Alexander Phase Detector.” Electronics Letters, vol. 52, no. 23, Institution of Engineering and Technology, Nov. 2016, pp. 1908–10. https://doi.org/10.1049/el.2016.3368

[19] Tertinek, Stefan, et al. “Binary Phase Detector Gain in Bang-Bang Phase-Locked Loops with DCO Jitter.” IEEE Transactions on Circuits and Systems Ii-express Briefs, vol. 57, no. 12, Institute of Electrical and Electronics Engineers, Dec. 2010, pp. 941–45. https://doi.org/10.1109/tcsii.2010.2083110

[20] “Error function”, https://en.wikipedia.org/wiki/Error_function

[21] N. Maghari, S. Kwon, G. C. Temes and U. Moon, "*Mixed-Order Sturdy MASH Δ-Σ Modulator*," 2007 IEEE International Symposium on Circuits and Systems, New Orleans, LA, USA, 2007, pp. 257-260, doi: 10.1109/ISCAS.2007.378384.

[22] “15-25 MHz Fractional-N Synthesizer”. www.aholme.co.uk/Frac2/Mash.htm

[23] Del Mar Hershenson, et al. “Design and Optimization of LC Oscillators.” International Conference on Computer Aided Design, 1999, pp. 65–69. https://doi.org/10.5555/339492.339574

[24] Aleksa, Martin et al. “Strategic R&D Programme on Technologies for Future Experiments,CERN-OPEN-2018-006”,Geneva,2018, https://ep-dep.web.cern.ch/sites/default/files/RD-ESPP-2018.pdf,doi: 10.17181/CERN.5PQI.KDL2

[25] Gardner, Floyd M. “*Phaselock Techniques*”. John Wiley and Sons, 2005.

[26] P. Roberts, “Understanding Phase Noise in RF and Microwave Calibration Applications”, 2008, NCSL International Workshop and Symposium, https://download.flukecal.com/pub/literature/9010168_ENG_A_W.pdf

[27] Clock and Data Recovery/Structures and Types of CDRs/the CDR Phase Comparator Wikibooks, Open Books for an Open World. en.wikibooks.org/wiki/Clock_and_Data_Recovery/Structures_and_types_of_CDRs/The_CDR_phase_comparator

[28] Leeson, D. B. (February 1966), “A Simple Model of Feedback Oscillator Noise Spectrum”, Proceedings of the IEEE, 54 (2): 329-330, doi:10.1109/PROC.1966.4682

[29] J.D.H. Alexander, "Clock Recovery from Random Binary Signals", vol. 11, pp. 541 - 542, October 1975 in Electronics Letters. © Institution of Electrical Engineers.

[30] Solomon, Jr, O M. PSD computations using Welch's method. [Power Spectral Density (PSD)]. United States: N. p., 1991. Web. doi:10.2172/5688766.

[31] “Bitwise Operator”, https://en.wikipedia.org/wiki/Bitwise_operation