IEEE Open Journal of
Circuits and Systems

Received 24 April 2023; revised 8 June 2023 and 30 June 2023; accepted 10 July 2023. Date of publication 14 July 2023; date of current version 4 August 2023.

Digital Object Identifier 10.1109/OJCAS.2023.3295649

Phase Interpolator-Based Clock and Data Recovery With Jitter Optimization

GEORGE SOULIOTIS[1] (Senior Member, IEEE), ANDREAS TSIMPOS[2,3] (Member, IEEE), AND SPYRIDON VLASSIS[2]

[1]Department of Electrical and Computer Engineering, University of Peloponnese, 263 34 Patras, Greece

[2]Department of Physics, University of Patras, 26504 Patras, Greece

[3]Adveos Microelectronics, 15451 Athens, Greece

This article was recommended by Associate Editor P.-H. Hsieh.

CORRESPONDING AUTHOR: G. SOULIOTIS (e-mail: gsoul@uop.gr)

The publication of the article was financially supported by HEAL-Link.

ABSTRACT In this paper, it is proposed a jitter analysis methodology, targeting on the optimization of a phase interpolator (PI) based clock and data recovery circuit (CDR). The methodology is applied for the optimized design of an 8-bit dual-loop CDR, designed with the CMOS TSMC 65 nm process node. The CDR is based on an extended, in terms of phase resolution, version, with a novel PI topology proposed in this work. The proposed CDR loop has a minimum frequency offset tracking ability equal to 500ppm at 5.83 Gbps, and so is suitable for adoption either in mesochronous or plesiochronous High Speed Serial Interface (HSSI) receivers. It consumes 14.2 mW with 1 V supply voltage and is able to achieve better than 10^{-10} Bit Error Rate (BER) performance. The CDR loop performance verification has been realized through the AMS simulator of Analog Design Environment of Cadence, by co-simulations of the transistor level CDR circuit with the Verilog-AMS based jitter generator.

INDEX TERMS Data acquisition, data communication, frequency synchronization, interpolation, phase locked loops, timing jitter.

I. INTRODUCTION

THE DEMAND for high data throughput in modern high-speed serial links continuously increases over the last few years. However, as data rates become greater, jitter on the transmitted signal, which is caused due to several external or internal noise sources, becomes a significant portion of the bit period. As a result, it undermines the transmission fidelity, making the signal recovery a very tough process for the receiver. Responsible for the task of data recovery on the receiver end, is the clock and data recovery circuit (CDR) [1], [2], [3], the accuracy of which in many cases determines the overall link performance.

The main task of a CDR circuit is to precisely recover the transmitted data sequence from the highly distorted receiver input signal. By detecting signal transitions, extracts the clock timing information from the received signal and generates a clock aligned at the center of the bit period. This clock is subsequently used to sample and recover the incoming bit stream with minimized skew and jitter [2], [3], [4]. The actual architecture of the CDR circuit is mainly determined by the transceiver clocking strategy, which may be a forward or embedded clock and in most cases is based on phase/frequency tracking feedback loops [1]. Feedback-less CDR architectures such as oversampling or gated-oscillator-based CDR have also been reported in the literature [5], mainly used in burst-mode optical applications due to their extremely fast acquisition rate. However, they suffer from decreased jitter suppression performance compared to their feedback loop-based counterparts [6], [7], [8], [9], [10]. Reference-less and frequency-detector-less CDRs can be substantially relaxed in hardware, but the jitter performance may be significantly reduced [11]. A transformer-coupled injection-locking CDR for high-speed optical networks is presented in [12], but an on chip transformer is required in this architecture occupying large area on the die. In the CDR loop, a large number of blocks

are involved, and each one contributes in a different but significant way to the noise performance. Thus, the systematic development of a CDR with optimized dynamic response should consist of a model including the noise contribution of the independent cells.

In this paper, a CMOS 8-bit dual-loop CDR with detailed design steps, focusing on the optimized dynamic response taking into account the noise contributions, is proposed. Also, a noise model, which is based on a phase interpolator (PI) topology is introduced. The proposed CDR loop has a minimum frequency offset tracking ability equal to 500 ppm at 5.83 Gbps, and so is suitable for adoption either in mesochronous or plesiochronous High Speed Serial Interface (HSSI) receivers. It consumes 14.2 mW with 1 V supply voltage and is able to achieve better than 10^{-10} bit error rate (BER) performance. The CDR loop performance verification has been realized through the AMS simulator of Analog Design environment of Cadence, by co-simulations of the transistor level CDR circuit with the Verilog-AMS based jitter generator proposed in [13].

The proposed CDR loop is based on a novel CMOS PI architecture which offers several advantages compared to the traditional CML-based phase interpolators such as: a) low power consumption b) simple inverter-based topology which can be easily extended to offer higher phase resolution c) ability to operate over a wide frequency range (by tuning the circuit time constants through a simple capacitor bank) while offers quite small phase error which is maintained almost constant over all the supported frequency range (CML PI architectures suffer from variable phase error over frequency) d) fast settling time e) quite simple control logic (can be implemented with simple thermo-coded registers) which simplifies the integration in the CDR loop and reduces the loop limit cycle effect which can lead in increased deterministic jitter due to dithering f) also due to its CMOS logic nature the circuit can be easily ported in several process g) capable for low power (low supply voltage) operation.

In this paper, the following are proposed a) a Jitter analysis methodology with Verilog-AMS-based jitter generator b) the design steps for a CDR according to the critical design considerations, and c) a new PI-based CDR with an improved resolution that validates the proposed methodology.

In what follows, an overview of the most commonly used feedback loop-based CDR architectures is given in Section II, while in Section III the main design considerations of PI-based CDRs are discussed. In Section IV, the new PI-based CDR topology is proposed, In Section V the CDR loop dynamics are analyzed and in Section VI simulation results with the proposed method are provided to verify the accuracy of the methodology.

II. CLOCK AND DATA RECOVERY OVERVIEW

The conceptual block diagram of a basic CDR loop is presented in Fig. 1. It consists of a retiming/sampling circuit, a Phase Detector (PD), a loop filter (LF), and a phase

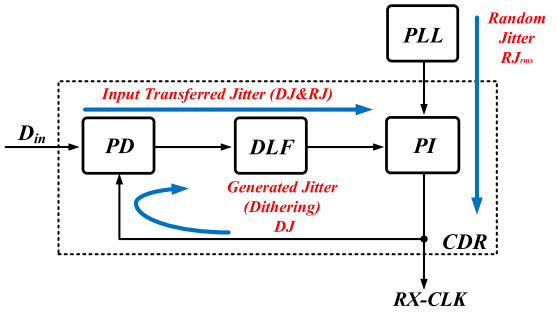

FIGURE 1. Basic CDR Loop block diagram.

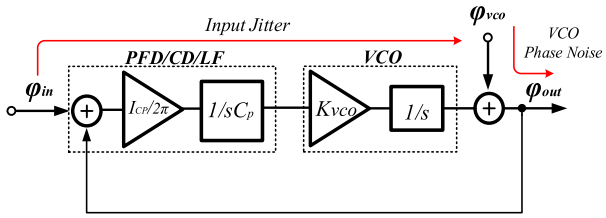

FIGURE 2. PLL-based CDR linear phase domain model.

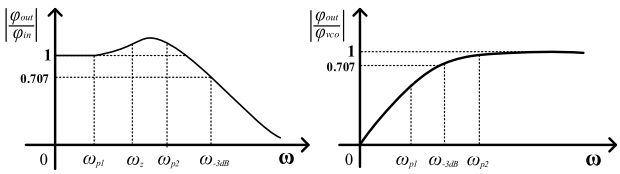

FIGURE 3. PLL-based CDR loop frequency response a) from CDR input to the CDR output b) from VCO to CDR output.

adjustment mechanism (Phase Adjustment), usually implemented with a phase interpolator (PI) [14], [15], [16], [17].

The architecture in Fig. 1 is very similar to a typical Phase Locked Loop (PLL) which also consists of a phase detector (PD), a loop filter (LF), and a VCO, providing the required phase adjustment mechanism [18]. However, PLL-based CDRs suffer from a serious drawback related to their loop bandwidth optimization, which becomes more significant at high data rates. In general, there are two main jitter sources in PLL-based CDRs. The first one is the jitter of the incoming signal, while the second one is VCO phase noise [2], [19]. The CDR ability to suppress these noise sources is determined by its loop bandwidth.

In Fig. 2, the linear phase domain model of the PLL-based CDR is presented. Due to its low pass transfer function of input phase (φ_{in}) to output phase (φ_{out}), the loop tends to track input phase excursions (jitter) of frequency lower than ω-3dB and to suppress phase variations of higher than ω-3dB frequency, as shown in Fig. 3a. On the other hand, the high pass response of VCO phase (φ_{vco}) to φ_{out} implies that the loop suppresses slowly varying phase excursions or noise of VCO clock and tracks high frequency variations, as shown in Fig. 3b. It is obvious that, in terms of noise suppression, there is a direct tradeoff between the input jitter and the

VCO phase noise. By selecting a narrow loop bandwidth, the input noise is suppressed but the VCO noise directly is passed to the output. On the other hand, a wide loop bandwidth suppresses the VCO noise, but the input jitter highly corrupts the CDR output clock.

The opposing constraints faced in loop BW optimization of PLL-based CDRs, raised the popularity of dual-loop CDR architectures [20]. By employing two independent feedback loops, these architectures can separate the process of RX clock synthesis, from the clock and data recovery process. Their main advantage is that the bandwidth of each loop can be independently selected, eliminating the trade-off in loop bandwidth optimization faced in PLL-based CDRs [14]. A wide bandwidth for the frequency synthesis loop can be used for better suppression of the VCO noise, while a narrow bandwidth in the phase/frequency tracking loop can offer the required input jitter rejection. In the dual-loop CDR architecture, a PLL is used to synthesize a clock f_{vco} at the same rate as the input data D_{in}, based on a low noise input reference clock f_{ref} coming from a crystal oscillator (XTAL). The PLL clock is subsequently applied to the DLL, which adjusting the delay of the voltage control line (VCDL), aligns the PLL clock at the edges of the incoming signal.

The phase/frequency tracking loop of DLL based CDRs, has a first order lowpass frequency response without requiring any extra zero in their transfer function for loop stability purposes, eliminating any closed loop transfer function peaking which is referred to as "jitter peaking". The main disadvantage of DLL-based CDRs is that due to the DLL limited phase capture range, they cannot tolerate any frequency offset between the transmitter and the receiver, which is a common case in embedded clock architectures [1], [21]. To improve the locking time of the DLL-based CDR the locking process can be divided into coarse and fine-tuning stages. During the coarse tuning stage, a self-starting control module is employed to eliminate the harmonic lock fast [22].

A widely used CDR architecture, with increased popularity in recent years, is the PI-based dual-loop CDR which is shown in Fig. 4, [1], [4], [6], [14]. It is an alternative implementation of the DLL-based CDR using a similar operation principle but employing a semi-digital DLL as a phase/frequency tracking loop instead of an analog one. The PI is employed as a phase adjustment mechanism instead of a VCDL, while the charge pump (CP) and the passive loop filter (PLF) are also replaced by a digital Finite State Machine (FSM) and a digital loop filter (DLF). Except for the PI which may be an analog block, the rest loop components are based on standard digital cells. The extended usage of digital components simplifies the design and offers increased stability over PVT variations [1]. Also, the adoption of a digital loop filter instead of an analog one eliminates the required large passive components and significantly reduces the required chip area.

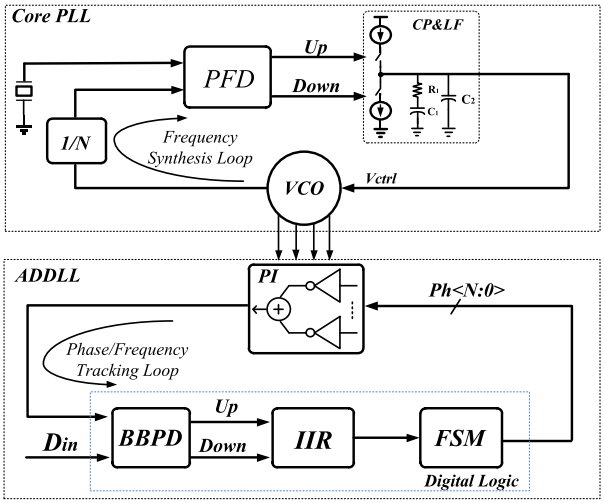

FIGURE 4. PI Based Dual oop CDR.

III. METHODS AND PROCEDURES, RESULTS

A. CDR LOOP DESIGN CONSIDERATIONS

Under real world operation conditions, there are several effects that undermine the clock and data recovery process. Timing errors caused due to jitter inherent both on the incoming signal and the recovered clock, frequency offsets between TX and RX clocking circuits, or even loop delays may significantly reduce the timing margins for proper sampling and increase the BER [3], [23], [24]. The CDR loop must be able to tolerate such non-idealities and recovers the incoming signal with high fidelity in order to maintain low BER.

One of the most critical performance specifications of a CDR loop is the so-called jitter tolerance. It depends on three parameters a) the CDR Loop bandwidth b) the PI phase resolution c) RX PLL clock rms Jitter. A wider CDR loop bandwidth increases the response of the loop in both slower or faster phase changes at its input (i.e., Jitter) increasing in this way the capability of the loop to track, tolerate and remain locked in the presence of higher amplitude slow or fast input jitter components. However, a wider loop bandwidth results in higher input jitter propagations at the CDR output clock. On the other hand, a lower phase resolution PI can also offer faster CDR loop response improving in this way the amplitude of Jitter that the CDR is able to tolerate in the expense of increased quantization error and Jitter generation. As a result, both the CDR loop BW and the PI phase resolution must be optimized in order to offer the required loop response and Jitter performance in order to ensure that the CDR is able to meet the Jitter tolerance performance dictated by the standard. The RX PLL clock Jitter should remain as low as possible since due to its random nature can increase substantially the RX clock peak-to-peak total jitter increasing in this way the system BER. A wider loop bandwidth for the PLL can offer better suppression of the VCO phase noise which is the main noise contributor improving in this way the PLL jitter performance. In PI based CDRs

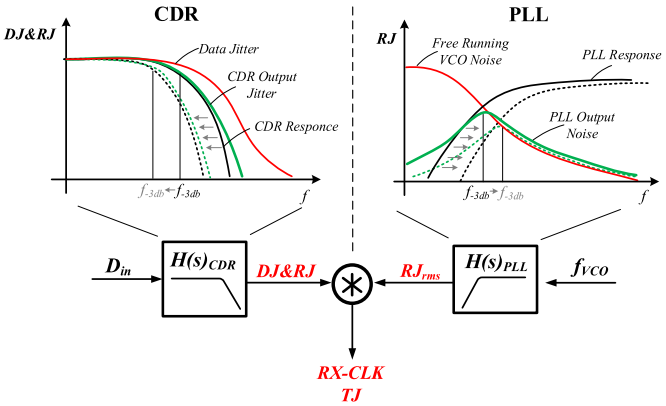

FIGURE 5. Dual-Loop CDR Jitter suppression by independently selecting each loop bandwidth.

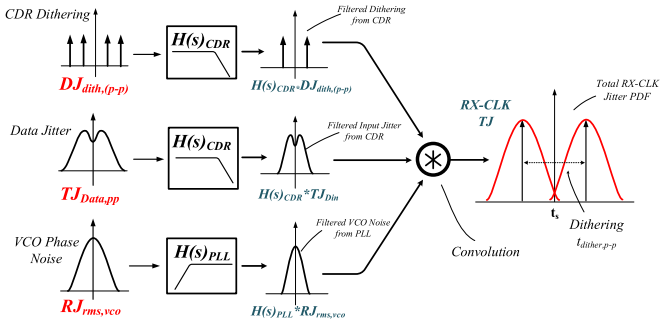

FIGURE 6. Resultant RX-CLK TJ PDF from the convolution of all jitter sources in the system.

the PLL and CDR BW can be optimized independently for better VCO noise suppression and lower input Jitter contribution accordingly which is one of main advantages of PI Based CDR architectures.

B. JITTER IN PI BASED CDRs

Jitter performance optimization is maybe the most challenging task during the CDR loop design procedure. During clock and data recovery, the amount of jitter which is inherent in both the incoming signal and the sampling clock (i.e., RX CLK), directly affects the CDR process reliability and degrades the RX BER performance. According to the additive jitter amount on the system, the timing margins may be significantly reduced, preventing the data recovery process reliability.

As illustrated in the block diagram of Fig. 1, there are three main Jitter sources in PI-based CDRs, which contribute to the RX-CLK total jitter. At first, RX-CLK is affected by the jitter transferred directly from the noisy input bit stream (e.g., Data Jitter). The amount of jitter passing from the CDR input to its output is determined by the loop BW. Another important jitter contributor on the RX-CLK, is the jitter generated by the CDR loop itself. Especially in PI-based CDRs, the main jitter mechanism responsible for the loop jitter generation is the so-called dithering, originating from the PI limited phase resolution which results in quantization error when the loop is in a steady state [3], [23], [25].

The contribution of each one of the aforementioned jitter sources in the RX-CLK total jitter (T_J), is determined by the PLL and the CDR loop dynamics. One of the most important advantages of Dual-loop CDR architectures is their ability to select independently the clock synthesis loop (e.g., PLL) bandwidth, from the phase/frequency tracking loop (i.e., CDR) bandwidth. Consequently, as illustrated in Fig. 5, a narrow CDR loop bandwidth can highly suppress input peak-to-peak Data Jitter (T_{Jpp}), while a wide PLL bandwidth significantly suppresses VCO phase noise and is able to provide a low random jitter *RJ* PLL clock. Eventually, the RX-CLK *TJ* PDF, can derive by convolving the aforementioned jitter sources PDFs, as illustrated in Fig. 6, [19], [20].

The probability of making wrong bit detections increases substantially in the presence of RX-CLK jitter, which displaces the sampling events from their ideal position and reduces the available timing margin. By convolving the RX-CLK *TJ* PDF_{TJ} with this of Data PDF_{TJ}, a Total Jitter PDF can be calculated, which includes the contribution of all the existing jitter sources in the system during the data recovery process. Dithering causes a shift equal to PI quantization error $t_{dither,p-p}/2$ (ideally ± 1 of PI step) of the *TJ* PDFs closer to each other, increasing in this way their overlap, and raises at the same time the probability for wrong bit detections. By taking into account the RX-CLK deterministic Jitter (*DJ*) and Random Jitter (*RJ*) components, the system BER can be estimated by Eq. (1) [19], where σ_{tot} is the overall *RJ* standard deviation value given by Eq. (2).

$$\mathrm{BER}(t) = p_T\left[\mathrm{erfc}\left(\frac{(t_c - t_{\mathrm{dith}}/2) - DJ_{pp}/2}{\sqrt{2}\sigma_{\mathrm{tot}}}\right) + \mathrm{erfc}\left(\frac{((t_c - t_{\mathrm{dith}}/2) - T) - DJ_{pp}/2}{\sqrt{2}\sigma_{\mathrm{tot}}}\right)\right] \quad (1)$$

$$\sigma_{tot} = \sqrt{\sigma_{data}^2 + \sigma_{clk}^2} \quad (2)$$

where, DJ_{pp} is the peak-to-peak value of *DJ*. So, it is evident that the BER performance of the RX highly depends on the CDR jitter suppression ability which directly calls for low loop bandwidth. However, there are several other CDR performance attributes, which make prohibitive the selection of an arbitrary narrow bandwidth and complicate the bandwidth optimization process.

C. QUANTIZATION ERROR

Due to PI quantized phase steps, the loop is not capable of completely eliminating phase errors between the RX local clock and the incoming data stream. Even in a locked state, there still exist small phase errors between RX-CLK and data, which force the phase detector to continually generate up/down indicators. As a result, in the locked state the PI continuously performs phase transitions and so the RX-CLK phase, shifts between adjacent phase steps.

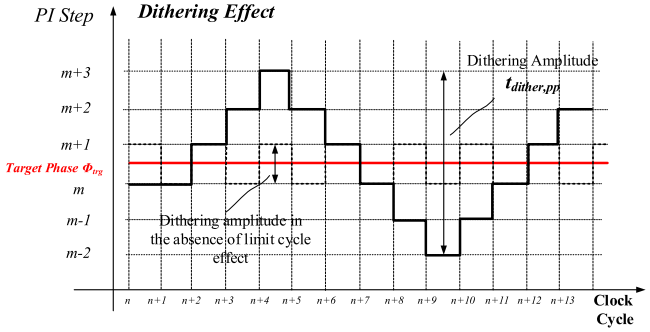

FIGURE 7. Locked state response of loop due to limit cycle effect.

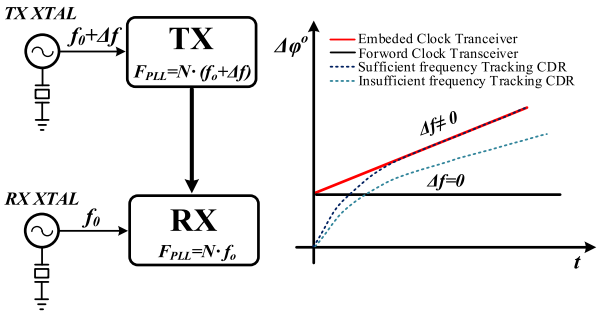

FIGURE 8. a) Plesiochronous transceiver with frequency offset. b) CDR phase offset tracking ability.

This effect is mainly referred to as dithering or quantization error and manifests itself as deterministic jitter on the RX-CLK. In RX-CLK jitter PDF, dithering is translated in a relocation of the RX clock crossing events from their mean value. The quantization error amplitude depends on PI phase resolution and its peak-to-peak value is equal to at least $\pm(1/2^{\mathrm{m}})\cdot$UI, as it alternates between two consecutive PI-steps (n and $n+1$ or n and $n-1$), where m is the PI resolution in number of bits.

However, in the presence of large loop delays, dithering may be larger than two PI phase steps. Generally, the locked state response of a bang-bang phase detector-based loop, such as a PI-based CDR, is a limit cycle whose amplitude and frequency are defined by the loop delays [23], [24]. Due to the limit cycle, a feedback decision requires a certain number of clock periods to take effect which also can increase the total quantization error amplitude of the loop. By considering both limit cycle and quantization error effects, an estimation for the loop dithering amplitude is given by,

$$t_{\mathrm{dither,pp}} = (2l+1)\cdot K_{\mathrm{PI}}\cdot K_{DLF} \tag{3}$$

where, l is the number of retiming stages in the loop, K_{PI} is the PI step in UI (i.e., PI Gain) and K_{DLF} is the loop filter gain [25].

Lower dithering amplitude can be achieved either by decreasing the PI gain K_{PI} (i.e., by increasing PI phase resolution) or by decreasing the DLF gain K_{DLF}. Both solutions lead to lower CDR loop bandwidth. As a result, a lower loop bandwidth is proven beneficial not only for the suppression of the incoming data stream jitter, but also for the loop jitter generation (Fig. 7).

D. CDR LOOP SLEW RATE

An also important characteristic of PI-Based CDR loops is their frequency offset tracking ability [3], [24], [25]. As illustrated in Fig. 8, in plesiochronous or embedded clock systems, both the receiver and transmitter employ an independent crystal oscillator (XTAL) for clock synthesis. In most cases there are small frequency offsets Δf, equal to some tens or hundredths parts per million (ppm), between the TX and RX XTALs, which are finally translated in a constant frequency offset Δf between TX and RX local clocks [3]. As a result, the data bit stream is clocked to a slightly different frequency compared to the RX clock, which leads in a slightly different phase accumulation rate ($\mathrm{d}\varphi = \Delta f * \mathrm{d}t$) between RX-CLK and D_{in}, as illustrated in Fig. 8. The CDR loop must be able to sufficiently track these frequency offsets in order to maintain locked, otherwise phase locking cannot be achieved leading the operation to fail and significantly increase the BER.

The frequency offsets tracking ability of a CDR loop is quantified by the loop slew rate LSR, which is an indicator of how fast the CDR is able to cover a phase distance equal to one UI and can be expressed as

$$SlewRate = \mathrm{TD}\cdot K_{\mathrm{PI}}\cdot K_{DLF} \tag{4}$$

where, TD is the data sequence transition density [1], [2]. As implied by Eq. (4), the slew rate is mainly determined by the PI phase resolution and the loop filter update rate (e.g., loop filter gain). In terms of loop dynamics, this is translated into a requirement for a wider loop bandwidth, which is in contrast with observations made in previous sections for jitter suppression.

Another critical characteristic of CDR which is closely related to the loop slew rate, is the time required from the loop to acquire phase locking at the beginning of operation. It is referred to as locking time and according to the application, a very fast initial locking may be necessary.

IV. PROPOSED PI-BASED CDR LOOP

The architecture of the proposed PI-based CDR loop is presented in Fig. 9. It consists of an 8-bit CMOS Phase interpolator (PI) [14], [15], [16], [26] a half-rate binary phase detector (HRBPD), a digital loop filter (DLF), a phase accumulator (PH-ACC) and a digital control unit (CDR CONTROL UNIT). The proposed CDR is the actual phase/frequency tracking loop of a dual-loop CDR architecture, intended to cooperate with a reference frequency synthesis loop. The frequency synthesis loop is a conventional analog Integer-N PLL, based on a differential RO-VCO which generates 8 reference phases of 45^o difference [27], [28].

Phase/frequency tracking is accomplished according to the algorithm presented in Fig. 10. The HRBPD compares

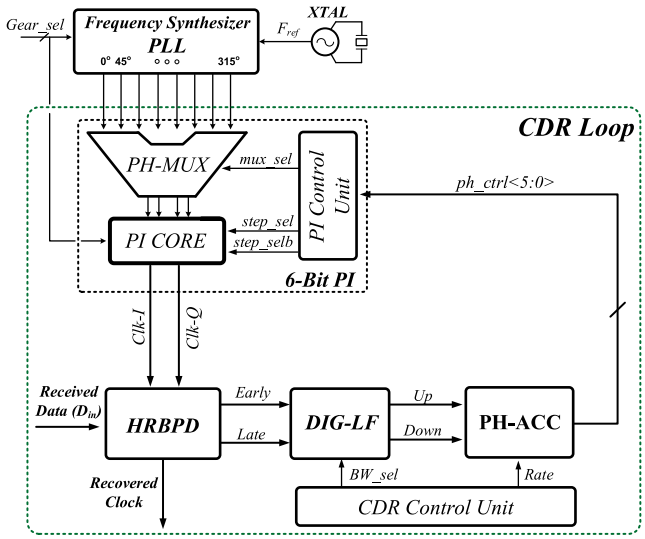

FIGURE 9. Proposed PI-based CDR architecture.

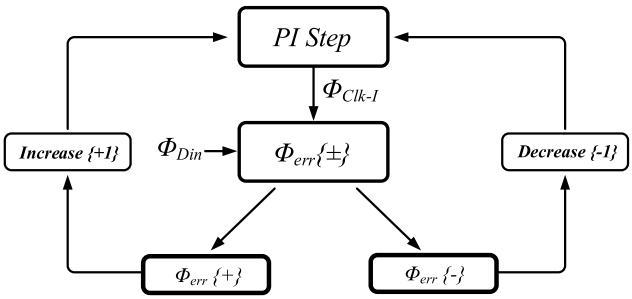

FIGURE 10. Phase/Frequency Tracking Algorithm of the proposed CDR.

the phase of the high-speed serial received data stream D_{in} with the in-phase interpolator clock output (*clk-I*), and according to their phase error Φ_{err} polarity $\{\pm\}$, generates early/late indicators when *clk-I* leads or lags the D_{in}. The DLF processes early/late indicators of HRBPD and generates up or down pulses only when they represent valid phase error information. The PH-ACC according to DLF requests (i.e., *UP* or *DN*), increases or decreases by one its current value and generates the 6-bit digital control signal <*ph_ctlr*>, which corresponds to one of the 64 PI phase steps. Finally, the PI CONTROL UNIT generates all the required control signals for the internal PI components and force PI to increase or decrease its output phase by one step. By following this sequence, the PI continuously shifts *clk-I* (in-phase clock) and *clk-Q* (quadrature clock) phases, until to align *clk-I* at the edges of the incoming data signal. After phase lock acquisition, the quadrature PI output *clk-Q*, which is aligned at the center of each bit period, samples the received bitstream D_{in} and delivers the recovered/retimed data, which is a delayed and less jittered version of D_{in}.

The 2-bit control signal <*gear*> adjusts the oscillation frequency of the PLL as well as the capacitor bank C_{bank} of the PI block [26] making feasible in different data rates. Finally, the CDR control unit generates the required control signals for the DLF and the PH-ACC.

The proposed architecture is generic and applicable among several different serial interface standards. In what follows, the main CDR sub-components and their characteristics are presented in more details.

A. PHASE INTERPOLATOR

The Phase Interpolator (PI) circuit [14], [16], [26] is maybe the most critical component of the CDR, since several important loop characteristics, such as Jitter generation, frequency tracking ability or the locking time highly depend on PI performance. As illustrated in Fig. 11, the PI [26] consists of two identical programmable PI-units, PI-unit I, and PI-unit Q, an 8x4 multiplexer (MUX) and an internal control unit (PI Control Unit), while simultaneously providing two orthogonal (Quadrature) clock outputs, *clk-I,* and *clk-Q* of 90^o phase difference, both required by the HRBPD. The MUX selects and provides the appropriate VCO output phases to PI-unit I and PI-unit Q. The schematic of MUX in transistor level is shown in Fig. 12. In the same figure the multiphase VCO (MPVCO) is shown along with the schematic of the delay cell.

Compared to [26] a phase resolution extension was achieved, by increasing the number of inverter pairs of each PI Unit to 8. Each inverter of the PI-Unit consists of a typical CMOS inverter with switches that allow to enable or disable the input signal to pass through the inverter. The circuit diagram is depicted in Fig. 13.

According to PI-Unit operation principle [26] a PI unit employing 8 pairs of equal weighted inverters, is able to provide 8 possible equidistant phase steps between two adjacent VCO phases, resulting to a phase resolution ($N_{PI-unit}$) equal to 3-Bit. Since the VCO phase resolution N_{VCO} is also 2^3 (8 phase steps) the PI total resolution is equal to:

$$N_{\mathrm{PI}} = N_{\mathrm{VCO}} \cdot N_{PI\_Unit} = 2^{(n+m)} = 2^6 \tag{5}$$

where, n and m are the VCO and PI phase resolutions in bits, respectively. Consequently, the PI of Fig. 12 is able to generate at total $2^6 = 64$ phases for each clock output, of $360^{\mathrm{o}}/64 = 5.625^{\mathrm{o}}$. The output phases $\theta^{\mathrm{o}}_{n,I}$ and $\theta^{\mathrm{o}}_{n,Q}$ of PI *clk-I* and *clk-Q* outputs can be described by the next equations:

$$\theta^{\mathrm{o}}_{n,I} = \theta^{\mathrm{o}}_{off} + \frac{n_t}{N_t} 360^{\mathrm{o}} \tag{6}$$

$$\theta^{\mathrm{o}}_{n,Q} = \theta^{\mathrm{o}}_{off} + \frac{n_t + 8}{N_t} 360^{\mathrm{o}} \tag{7}$$

where, N_t and n_t are integers, $N_t = 64$ is the total phase steps between 0^{o} and 360^{o} and n_t ranges between 0 and 63 and defines the specific phase step and is the internal phase offset of PI.

B. HALF RATE BINARY PHASE DETECTOR

The half rate binary phase detector (HRBPD) topology is a conventional bang-bang, Alexander type half-rate PD, which is based on the early-late detection method [29], [30], [31]. The circuit diagram is shown in Fig. 14(a), while in

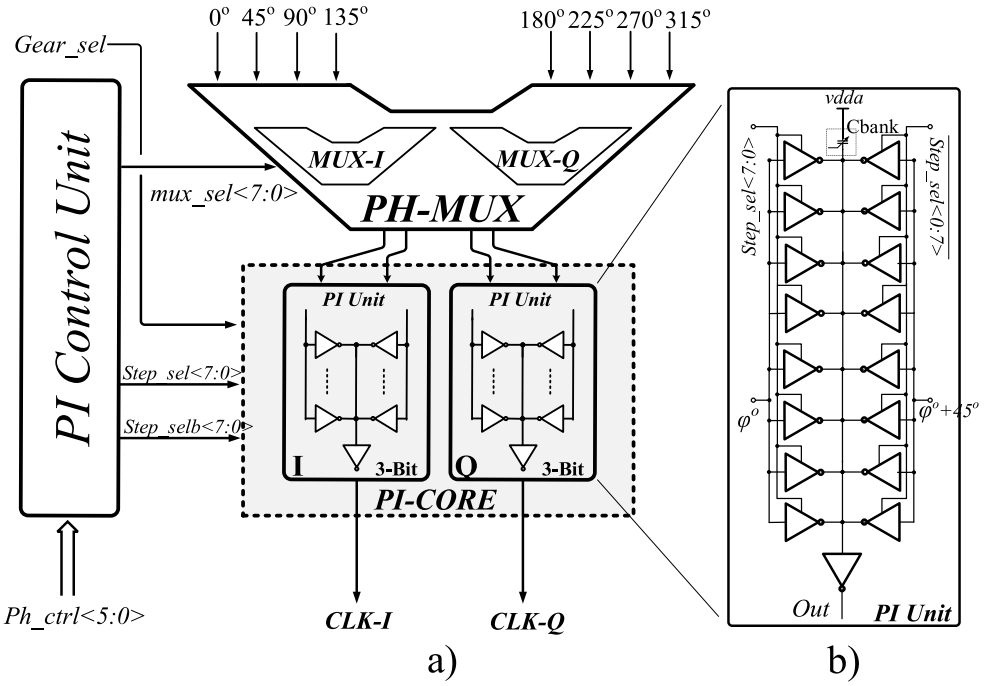

FIGURE 11. a) 6-Bit Phase Interpolator architecture. b) PI-Unit topology.

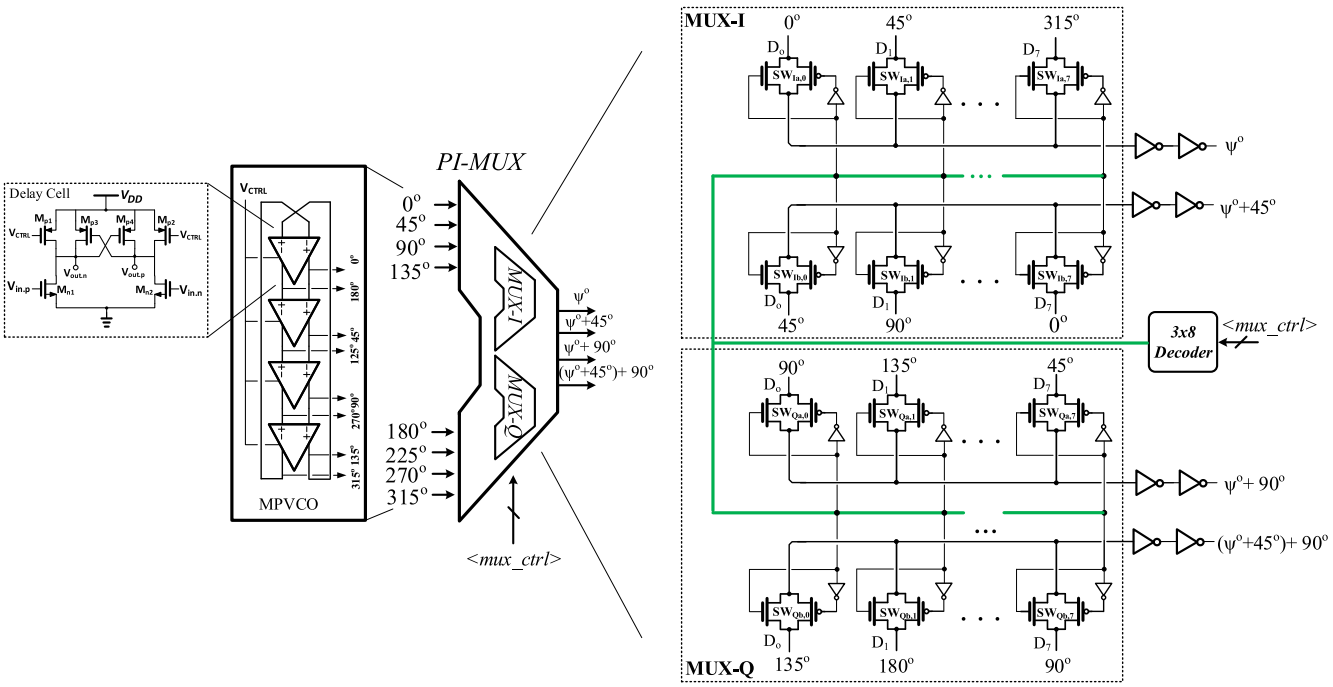

FIGURE 12. a) 6-Bit Phase Interpolator architecture. b) PI-Unit topology.

Fig. 14(b) the phase detection methodology is presented. PD samples the received data bitstream and determines when there is a data transition and when *clk-I* leads or lags compared to D_{in} [2], by using both the in phase (*clk-I*) and the quadrature clock phases (*clk-Q*) of the PI. D_{in} is sampled by both rising and falling edges of *clk-I*, while from *clk-Q* only by its rising edge.

Due to noise and circuits non-idealities under real operation conditions, the PD transfer function is effectively linearized and presents a finite gain. For instance, for a small phase difference between D_{in} and *clk-I*, the PD flip flops are incapable to reach their full output levels due to metastability effect generating as result invalid up or down indicators. To minimize metastability effect, dynamic D flip-flops (DFF)

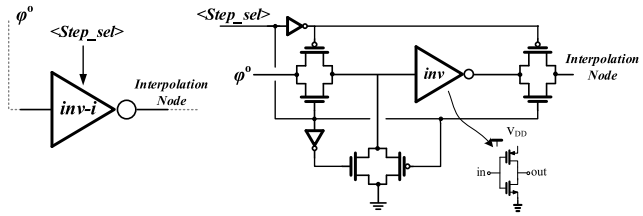

FIGURE 13. PI-Unit Inverter circuit.

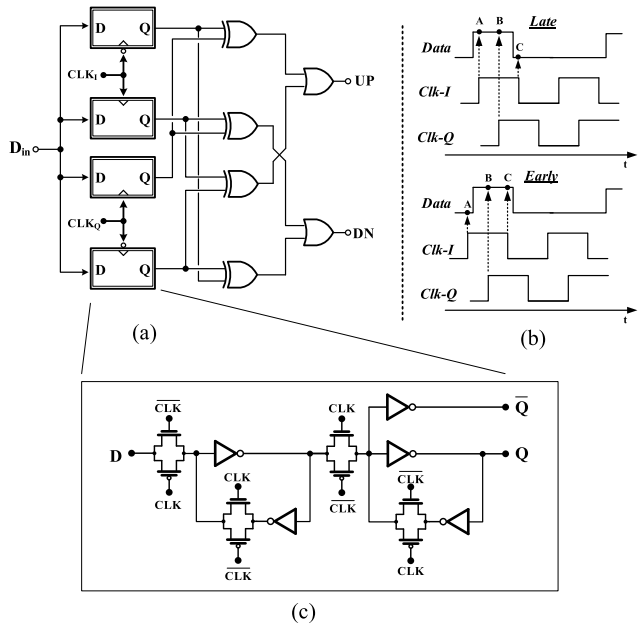

FIGURE 14. Half rate bang-bang phase detector a) circuitry topology b) phase detection methodology and c) DFF circuit.

have been employed in the BBPD, as shown in Fig. 14(c). Also, the design of DFF is fully customized and optimized to cover the required operating frequency.

A widely used approximation in the literature [3], [30], [31], [32] for the phase detector transfer function linearization, is that the BBPD characteristic can be effectively linearized by taking into account the jitter of the input data signal. Assuming that the data jitter has a Gaussian PDF with standard deviation σ, the effective transfer function derives through convolution of the jitter PDF with the PD ideal characteristic. Since a larger number of Jitter samples are included in the range $-2\sigma\ \tau o\ 2\sigma$, it can be finally said that the PD characteristic exhibits linear gain in this range, given by Eq. (8) where, $TJ_{pp} = 2\sigma/\text{UI}$ is the amplitude of data jitter.

$$K_{PD} = \frac{2}{TJ_{pp}} = \frac{\text{UI}}{\sigma}. \tag{8}$$

C. DIGITAL LOOP FILTER AND PHASE ACCUMULATOR

Even in locked state, the PD continually executes phase error estimation and generates early/late indicators in every clock cycle. However, due to jitter inherent on the input signal and the RX clock, or flip flops metastability, some indicators do not provide valid phase error information. As a result, they must be ignored to avoid excessive jitter generation due to PI phase shifts. For this purpose, a moving average Digital Loop filter (DLF), which processes the early/late indicators of HRBPD and generates an up or down signal only when a valid phase error information is represented has been used.

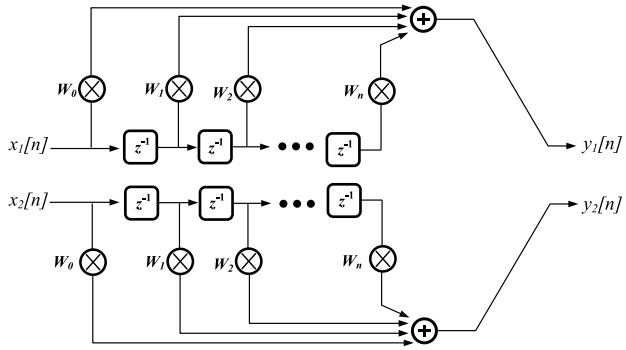

FIGURE 15. DIG-LF block diagram.

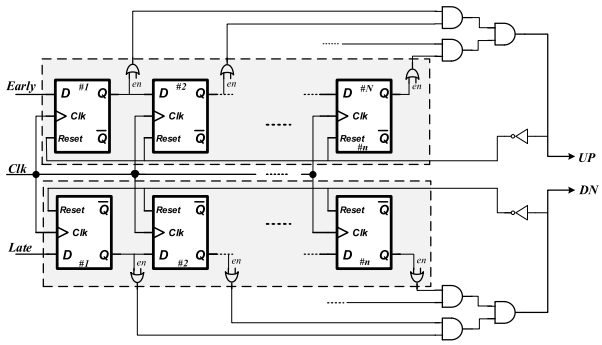

FIGURE 16. DIG-LF circuit.

The block diagram and the circuit of DLF are presented in Fig. 15 and Fig. 16, respectively. The flip-flop topology is the same as those used for the design of PD and presented in Fig. 14(c). It is a n-tap, moving average variable FIR filter, implemented by two n-bit Serial Input-Parallel Output shift registers, followed by two combination logic-based summers. The filter splits into two paths. The upper path processes early indicator which comes from HRBPD while the lower path processes late indicators. The shift registers store early/late indicators and generate an up or down pulse for the PH-ACC, only when n consecutive identical indicators are observed at the input.

In Fig. 15, $x_1[n]$ and $x_2[n]$ are denoted as the input signals of DLF, namely the early or late indicators coming from PD, while $y_1[n]$ and $y_2[n]$ are the output signals of DLF, which make the final decision for the early or late control for PI. The weight factors of DLF w_0 to w_n are equal to one in this case. The length n of the shift registers is variable in the range $n = [0, 8]$, offering the benefit of adaptive loop bandwidth. Each output $y[n]$ of the DLF is described by Eq. (9) while the loop filter transfer in z-domain is given by Eq. (10)

$$y[n] = \sum_{k=0}^{N-1} h[k]x[n-k] \tag{9}$$

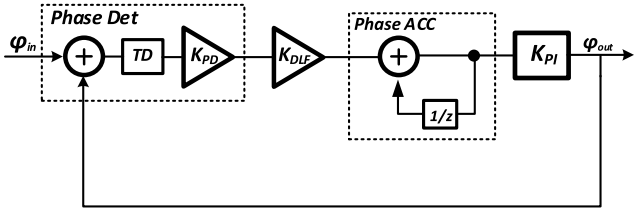

FIGURE 17. CDR model in z-domain.

$$Y(z) = \left(\frac{1}{N}\sum_{k=1}^{N-1} z^{-k}\right)X(z) \tag{10}$$

where, N is the filter taps (i.e., the shift registers length in bits) while $h[k] = 1/(N+1)$ is the filter coefficients sequence. A phase accumulator is used which generates the 6-bit phase step control word *ph_ctrl<5:0>* producing the suitable value corresponding to one of the 64 PI steps. It is a 6-bit synchronous up/down counter. The counter stores the value of the current phase step, and according to the up or down indicator coming from the loop filter increases or decreases its content by one. Due to the fact that the output of the counter is the weighted sum of its current content with the incoming up/down indicators it can be considered as an integrator in z-domain, having a transfer function given by,

$$H(z) = \frac{1}{1 - z^{-1}} \tag{11}$$

where, T_s is the accumulator/counter clock period.

V. CDR LOOP DYNAMICS

The z-domain model of the proposed CDR in which all the loop sub-components have been replaced with their equivalent in z-domain is depicted in Fig. 17.

As already explained, the HRBPD characteristic can be effectively linearized, having a gain K_{PD} which is approximated by Eq. (8). However, due to the fact that the number of comparisons that the phase detector realizes depends on number of data sequence transitions, an extra constant gain representing the transition density T_D is inserted in the z-domain model, and the total P_D gain is finally assumed equal to

$$K_{PD,tot} = TD \cdot K_{PD} \tag{12}$$

The DLF consists of only a proportional path, having a transfer function given by Eq. (10) as already mentioned. However, for low frequencies well below the digital cells clock frequency it can be approximated by a constant gain coefficient which is equal to

$$K_{DLF} = \frac{f_{DLF}}{N} = \frac{1}{N \cdot T_{DLF}} \tag{13}$$

where, N is the length of the DIG-LF shift register, while T_{DLF} is the loop filter clock period. The K_{PI} represents the ability of the loop to divide the clock cycle in a number equally spaced discrete phases and so in z-domain it can be represented by a constant gain given by,

$$K_{\mathrm{PI}} = \frac{T_{per}}{2^m} = \frac{2UI}{2^m} \tag{14}$$

where, T_{per} is the period of the RX-CLK (i.e., the PLL clock period) and m is the PI phase resolution in number of bits. Finally, as already explained, the accumulator in z-domain can be treated as an integrator having a transfer function given by Eq. (11).

Based on the aforementioned analysis and the z-domain model of Fig. 17, the open loop and the closed loop transfer functions of the CDR can be derived, as given by Eq. (15) and Eq. (16), respectively.

$$H(z)_{open} = TD \cdot K_{PD} \cdot K_{PI} \cdot K_{DLF} \cdot \left(\frac{1}{1 - z^{-1}}\right) \tag{15}$$

$$\begin{aligned} H(z)_{closed} &= \frac{\varphi_{out}}{\varphi_{in}} = \frac{H(z)_{open}}{1 + H(z)_{open}} \\ &= \frac{TD \cdot K_{PD} \cdot K_{PI} \cdot K_{DLF}\left(\frac{1}{1-z^{-1}}\right)}{1 + TD \cdot K_{PD} \cdot K_{PI} \cdot K_{DLF}\left(\frac{1}{1-z^{-1}}\right)} \\ &= \frac{TD \cdot K_{PD} \cdot K_{PI} \cdot K_{DLF}}{-z^{-1} + TD \cdot K_{PD} \cdot K_{PI} \cdot K_{DLF} + 1} \end{aligned} \tag{16}$$

The CDR closed loop transfer function, can be also transformed from z-domain to s-domain, by using the backward Euler method [32], according to which:

$$z = e^{-sTs} \approx 1 - sT_s \tag{17}$$

where, T_s is the sampling clock period. For the specific case of the proposed CDR, T_s is the synchronous up/down counter clock period denoted as T_{acc}. By substituting Eq. (17) to Eq. (16), the s-domain closed loop transfer function is given by Eq. (18) derives.

$$\begin{aligned} H(s) &= \frac{\varphi_{\mathrm{out}}(s)}{\varphi_{\mathrm{in}}(s)} = \frac{K_{PI} \cdot (TD \cdot K_{PD}) \cdot K_{DLF}/T_{acc}}{s + K_{PI} \cdot (TD \cdot K_{PD}) \cdot K_{DLF}/T_{acc}} \\ &= \frac{\omega_{-3db}}{s + \omega_{-3db}} \end{aligned} \tag{18}$$

$$f_{-3db} = \frac{K_{\mathrm{PI}} \cdot (TD \cdot K_{PD}) \cdot K_{DLF}}{2 \cdot \pi \cdot T_{acc}} \tag{19}$$

As implied by Eq. (18) the CDR loop, is a first order system exhibiting a low pass characteristic from φ_{in} to φ_{out} with a cut-off frequency f_{-3dB}, given by Eq. (19). By replacing in Eq. (19) each CDR loop component gain coefficient, K_{PI}, K_{PD} and K_{DLF}, with the corresponding quantities given by Eq. (14), Eq. (8) and Eq. (13) respectively, a more comprehensible in terms of loop parameters expression for the loop cut off frequency can be derived

$$f_{-3db} \sim \frac{2 \cdot \mathrm{TD}}{2^m \cdot TJ_{pp} \cdot N \cdot \pi} \tag{20}$$

As expected, the loop bandwidth is inversely proportional to the PI phase resolution m and the loop filter shift register length N. However, the most important observation is that the CDR bandwidth highly depends on the Data sequence

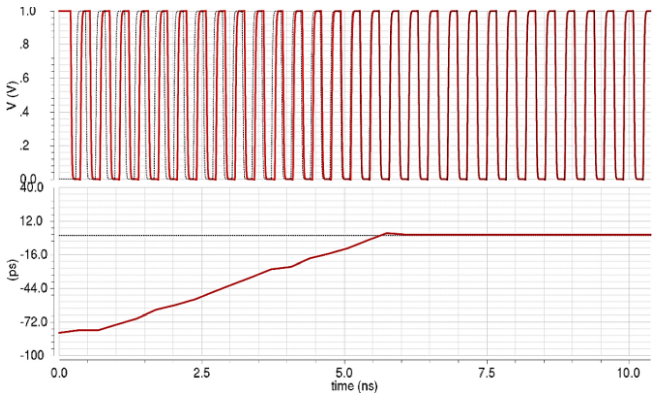

FIGURE 18. a) *clk-I* and D_{in} signal during phase acquisition process. b) Time delay between *clk-I* and D_{in}.

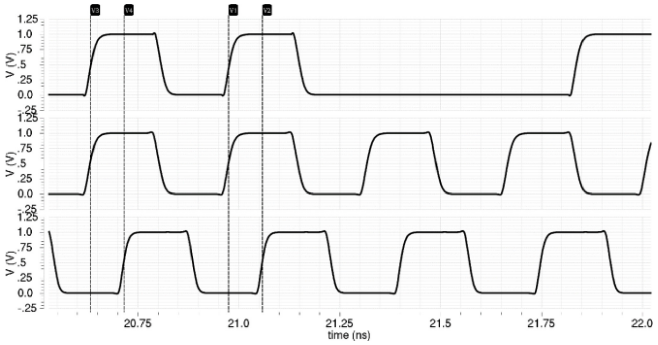

FIGURE 19. CDR loop transient response in locked state.

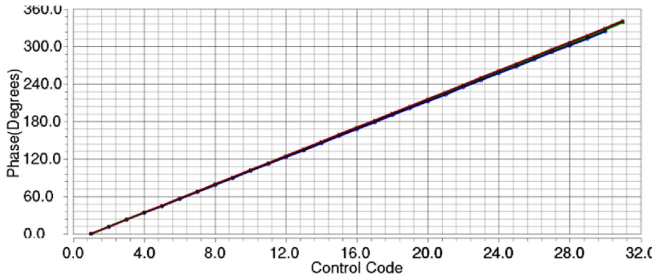

FIGURE 20. PI linearity vs the control code for 1.5/3/6 Gbps.

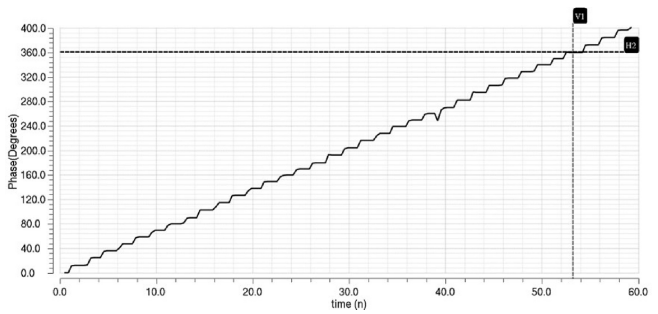

FIGURE 21. Phase steps transitions from 0 to 360°.

transition density and the loop Jitter. This is a common problem faced in bang-bang phase detector-based loops, such as all-digital PLLs or all-digital CDRs, that complicates a lot the precise estimation of the loop bandwidth through a behavioural modelling.

VI. RESULTS

The proposed CDR loop has been developed in transistor level and designed in TSMC 65nm technology node. Simulations were performed for operation at 5.83 Gbps with Spectre and AMS simulators of the Analog Design Environment of Cadence, under 1.0 V supply voltage. In order to confirm its ability to operate in real world HSSI applications, the M-PHY standard ver.3.0 specifications have been utilized as a reference point in several test cases. The circuit's performance verification has been realized mainly through co-simulations of the Verilog-AMS based Jitter generator proposed in [13], and the CDR top level schematics. The high parameterization of the Jitter generator offers the ability to easily generate all the required stress input signals for the verification of several important performance metrics such as the CDR Jitter Tolerance and Jitter transfer.

In Fig. 18(a), with red line the PI in-phase clock output (*clk-I*) and with black line the CDR input bitstream D_{in} is depicted, during the phase acquisition process. In Fig. 18(b) with red line the time delay between the two signals is illustrated. As it is shown, at initial point *clk-I* leads D_{in} by 75 ps which corresponds to 78°, however after 5.5 ns (i.e., 18 clock cycles) the loop acquires lock and eliminates the phase error between the two signals. In this case, for illustration purposes, the transition density of D_{in} was set equal to $TD = 1$ and so the aforementioned phase acquisition times are not representative for real world operation conditions.

In Fig. 19 the CDR random input data signal as well as the in-phase (*clk-I*) and quadrature (*clk-Q*) clock outputs of the PI for operation in locked state are presented. In locked state, *clk-I* is aligned at the edges of the incoming signal while *clk-Q* is aligned at the center of the data eye.

As mentioned in Section III-C, one of the main Jitter contributors in PI-based CDRs is the loop Jitter generation due to PI quantization error or large loop delays. The loop jitter generation has been tested for several DLF settings (i.e., shift register length) under the presence of the totally jitter free input signal.

Known problems in PI-based CDRs is the PI linearity versus the step control and the possible glitches appeared in output during the phase change. The glitches are stronger when the phase change takes place on the multiples of 90° of input phase [26]. The first problem can be solved by selecting the capacitor value to create the corresponding time constant, while the second by allowing the phase change to occur in the middle of the clock pulse width. The linearity of the PI as function of the control code is shown in Fig. 20 for three values of bit rates, namely 1.5/3/6 Gbps. The maximum phase error is around 1° over PVT corners and over the bit rates [26]. The transient response during phase change is shown in Fig. 21. The results in both figures are based on postlayout simulation results and show that the mentioned

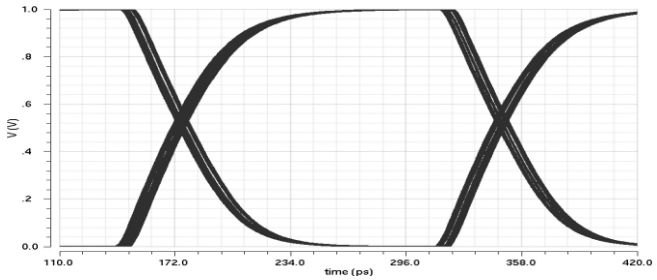

FIGURE 22. Recovered Clock eye Diagram for operation with a 4-Bit Loop Filter.

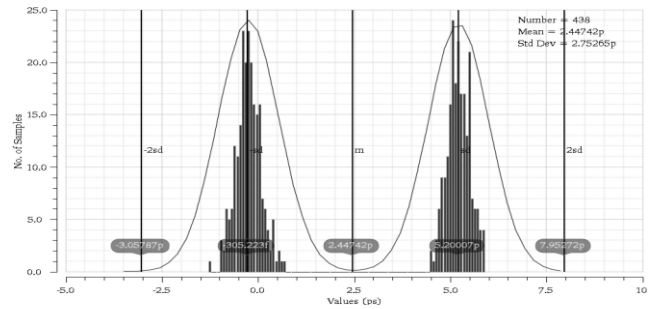

FIGURE 23. Recovered Clock Jitter Histogram for operation with a 4-bit Loop Filter.

TABLE 1. CDR loop jitter generation for several loop filters length.

Filter	Dithering Steps	DJ_{pp}
1-Bit	6	~26.8ps
2-Bit	4	~16.1ps
3-Bit	2	~5.4ps
4-Bit	2	~5.4ps
5-Bit	2	~5.4ps
6-Bit	2	~5.4ps
7-Bit	2	~5.4ps
8-Bit	2	~5.4ps

problems of non-linearity and glitches during step transitions have been minimized.

The eye diagram of the CDR recovered clock for operation with a 4-bit loop filter is presented in Fig. 22 showing, in this case, dithering amplitude equal to ±1 steps. The corresponding jitter histogram (Fig. 23) confirms that the recovered clock alters by two steps around the target phase, resulting in a jitter generation amplitude equal to 11 ps. Jitter generation simulation results for operation at 5.83 Gbps for several loop filters settings are summarized in Table 1. As is shown for 1-Bit and 2-Bit loop filters, the limit cycle is the effect that mainly determines the dithering amplitude which is equal to 6 PI steps and 4 PI steps, respectively. For loop filters higher than 3-Bits the limit cycle effect is eliminated, and the dithering amplitude becomes equal to 2 PI steps. The results confirm that a lower loop filter gain and thus a lower loop bandwidth attenuates the loop jitter generation.

Maybe the most important performance metric of a CDR loop is the Jitter Tolerance. To test the Jitter tolerance performance of the CDR Loop at the first maximum available timing margin, (or else the maximum acceptable phase error Φ_{err} between the Data signal and the CDR clock) for errorless operation must be calculated. The M-PHY standard specifies that the target BER performance for the system is 10^{-10} which corresponds to a QBER $=$ 6.36, while the worst case input (peak to peak total Jitter) TJ_{pp} that the receiver must be able to tolerate is equal to 0.52 UI. It further clarifies for the input signal that DJ_{pp} $=$ 0.35 UI and RJ_{pp} $=$ 0.17 UI (i.e., $RJ_{rms,data}$ $=$ 0.027 UI).

To make accurate the estimation of t_{margin} value, the Random Jitter of the RX-PLL ($RJ_{rms,rxclk}$) must be included in the calculations, where t_{margin} is half of the time opening of the data eye diagram. According to the input jitter estimation methodology presented in Section III-B, the RX-CLK random jitter can be taken into account in the evaluation of the TJ_{pp} value for a specific BER, as implied by Eq. (1). The TJ_{pp} value can be approximated by,

$$\begin{aligned} \mathrm{TJ}_{pp} &= \mathrm{DJ}_{pp} + Q_{BER} \cdot \sqrt{\sigma_{data} + \sigma_{rx,clk}} \\ &= \mathrm{DJ}_{pp} + Q_{BER} \cdot \sqrt{\mathrm{RJ}_{rms,data} + \mathrm{RJ}_{rms,rxclk}} \end{aligned} \quad (21)$$

in which the RX-PLL noise contribution is also taken into account.

Owing to the fact that the design of the RX-PLL is outside the objectives of this work, there is not available an explicit value for the $RJ_{rms,rxclk}$ and so, a realistic approximation of it must be realized. According to several published works in the literature [33], [34], [35], the design of a PLL suitable for HSSIs RX applications operating near the required frequency (e.g., around 3 GHz), of even lower than 0.003 UI or 0.5 ps rms RJ_{rms} is feasible. However, in order to relax the requirements for the RX-PLL block, we assumed a $RJ_{rmx,rxclk}$ value equal to 1 ps or 0.006 UI which corresponds in a t_{margin} equal to 0.235 UI. Intending to further ensure the correctness of the Jitter tolerance test, the limit of the maximum allowable phase error between data signal and the CDR clock, has been finally specified equal to 0.2 UI. Even though this is a pessimistic estimation, it gives even more margin for potential unpredictable non-idealities which may exist in the system under real world operation condition and affect the data recovery process.

The testbench configuration used for the verification of the CDR Jitter tolerance is illustrated in Fig. 24. It includes the CDR block in schematic level, the proposed in [13] Jitter Generator, and a clock generator described in Verilog ams language. The Jitter generator has been used to generate and modulate the CDR input signal phase, with sinusoidal jitter of a certain frequency F_{SJ} and amplitude A_{SJ} each time. The clock generator mimics the operation of the RX local clock synthesizer (RX-PLL), and generates a multiphase clock of 2.915 GHz frequency and 1 ps RJ_{rms} for the CDR PI.

The jitter tolerance test has been performed based on the methodology described in [13]. More specifically a random bit sequence whose phase was modulated by a different sinusoidal jitter frequency F_{SJ} was applied at the CDR input each time, and the phase error Φ_{err} between the CDR clock (i.e., *clk-I*) and the input signal D_{in} was measured in locked state. For a specific sinusoidal jitter FSJ the jitter amplitude

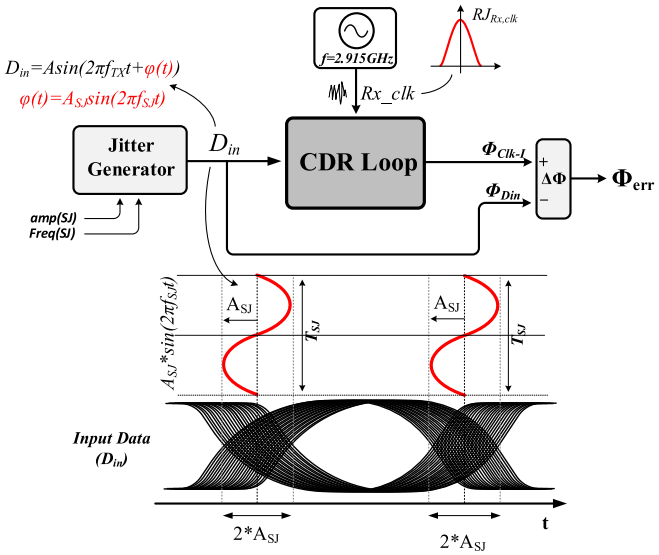

FIGURE 24. Jitter Tolerance performance verification Testbench.

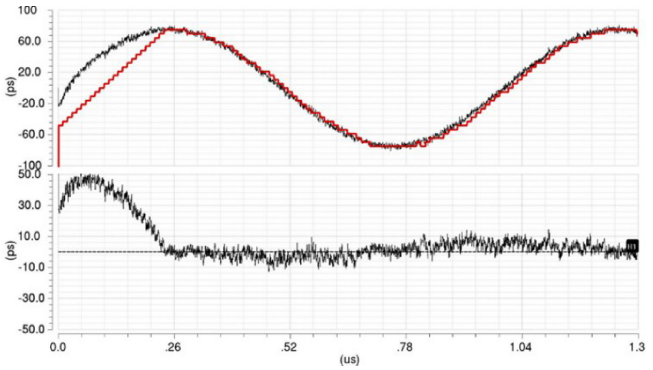

FIGURE 25. Jitter tolerance simulation with 8-Bit DIG-LF for fSJ = 1.0MHz and ASJ = 0.44UI a) D_{in} phase (black line) versus *clk-I* phase (red line) b) phase error between D_{in} and *clk-I*.

A_{SJ} was increased until a larger than 0.2 UI phase error between *clk-I* and D_{in} was observed at the CDR output. Afterwards, the F_{SJ} was adjusted to another value and the test was repeated. The minimum sinusoidal Jitter modulation frequency F_{SJ} used was equal to 0.3 MHz while the maximum 12 MHz. Tests were performed by using 6-Bit as well as 8-Bit CDR loop filters, in order to investigate the impact of loop bandwidth on the Jitter tolerance performance of the system.

Several test iterations for operation with 6-Bit and 8-Bit loop filters (Dig-LF) under the presence of various SJ modulation frequencies and amplitudes have been conducted for operation at 5.83 Gbps.

Simulation results for operation with 8-Bit Dig-LF and for sinusoidal input Jitter modulation with $F_{SJ} = 1$ MHz frequency and $A_{SJ} = 75$ ps (0.44 UI). This case is one of the worst and therefore the phase error diagram are presented in Fig. 25(b) together with the D_{in} phase versus *clk-I* phase, in Fig. 25(a). As shown from the figures, the maximum phase error between *clk-I* and D_{in} observed is lower than 20 ps or 0.116 UI.

TABLE 2. CDR loop jitter generation for several loop filters length.

SJ Modulation Freq F_{SJ}	CDR Maximum Tolerable SJ Amplitude A_{SJ} (UIpp)	
	6-Bit DIG-LF	8-Bit DIG-LF
0.3MHz	4.9 UI	3.2UI
0.6MHz	3.1UI	1.6UI
1MHz	1.7UI	1UI
2MHz	1.2UI	0.6UI
4MHz	0.6UI	0.3UI
8MHz	0.4UI	0.25UI
12MHz	0.34UI	0.25UI

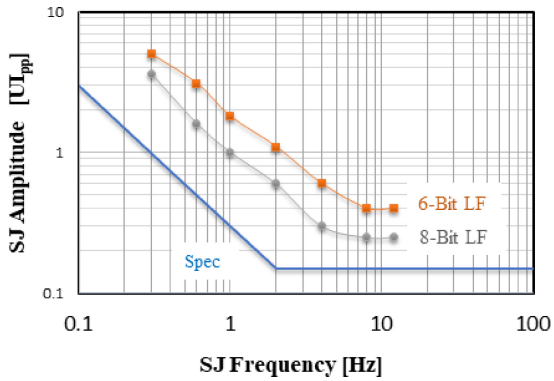

FIGURE 26. CDR Jitter tolerance curves at 5.83Gbps for operation with 6-Bit (red line) and 8-Bit (grey line) loop filter vs MPHY standard spec (blue line).

Although not shown in detail in this paper, some other cases have been simulated with the 6-Bit Dig-LF. One case is for input signal phase modulation with sinusoidal jitter $F_{SJ} = 0.3$ MHz and $A_{SJ} = 400$ ps (2.3 UI). After the phase locking, the phase error between the two signals has been proved lower than 15 ps or 0.09 UI, which is lower than the specified Φ_{err} limit. Therefore, it can be said that the CDR tolerates the input Jitter properly. A second case is for input signal phase modulation with sinusoidal jitter $F_{SJ} = 1$ MHz and $A_{SJ} = 150$ ps (0.9 UI). The maximum observed phase error in locked state is lower than 20 ps (0.116 UI) which is lower than 0.2 UI indicating that the loop does not exceed the target BER and it is again able to tolerate the input jitter.

The final results derived from several test iterations for operation with 6-Bit and 8-Bit loop filters under the presence of various SJ modulation frequencies and amplitudes are summarized in Table 2 while the derived Jitter tolerance curves for operation at 5.86 Gbps with 6-Bit and 8-Bit loop filters are presented in Fig. 26 together with the curve defined by MPHY spec. As expected for lower loop bandwidth (i.e., higher loop filter gain) the CDR is able to tolerate less input Jitter. However, in both cases, the CDR exceeds the M-PHY standard specification and so according to simulation results, the loop is able to meet the target BER performance (i.e., 10^{-10}).

Another important performance metric of the CDR loop is the so-called Jitter transfer. Jitter transfer quantifies the

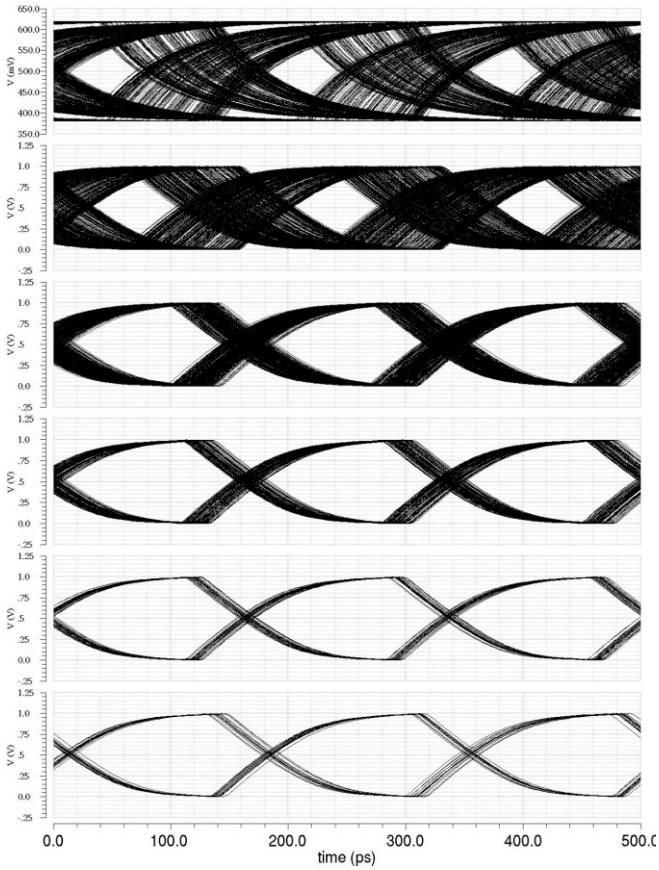

FIGURE 27. From top to bottom a) Input signal eye diagram modulated by sinusoidal Jitter. b) Recovered clock for 2.5MHz sinusoidal jitter. c) Recovered clock for 5MHz sinusoidal jitter. d) Recovered clock for 7.5MHz sinusoidal jitter. e) Recovered clock for 10MHz sinusoidal jitter. f) Recovered clock for 15MHz sinusoidal jitter.

amount of Jitter which is transferred from the CDR input to the CDR output (i.e., CDR clock) and actually represents the closed loop transfer function of the system. The CDR loop Jitter Transfer has been tested by using again the test-bench configuration of Fig. 24. In this case multiple test iterations have been performed by applying a random input stress test signal, modulated by sinusoidal jitter of constant modulation amplitude equal to 0.55 UI but different SJ modulation frequency each time. The SJ modulation frequency was varied from 1 MHz to 20 MHz, in steps of 2.5 MHz for modulation frequencies until 10 MHz and in steps of 5 MHz for higher modulation frequencies. In order to quantify the Jitter amount which is transferred from the CDR input to the CDR output, the recovered clock peak to peak recovery jitter was measured in each case (i.e., for different SJ modulation frequencies), by plotting the eye diagram of the CDR in-phase clock output (*clk-I*).

In Fig. 27(a) is illustrated the modulated by SJ input stress test signal used in Jitter transfer simulation in case of performing with an 8-Bit Loop filter. In Fig. 27(b, c, d, e and f) are illustrated the eye diagrams of the recovered clock (*clk-I*) for SJ modulation frequencies of 2.5 MHz, 5 MHz, 7.5 MHz, 10 MHz and 15 MHz, respectively, showing that

TABLE 3. Jitter transfer performance summary.

Jitter Modulation Frequency	Jitter Modulation Amplitude	Output Jitter (6-Bit LF)	Output Jitter (8-Bit LF)
1MHz	0.58UI	0.58UI	0.58UI
2.5MHz	0.58UI	0.58UI	0.55UI
5MHz	0.58UI	0.47UI	0.3UI
7.5MHz	0.58UI	0.35UI	0.195UI
10MHz	0.58UI	0.26UI	0.11UI
15MHz	0.58UI	0.18UI	0.08UI
20MHz	0.58UI	0.1UI	0.05UI

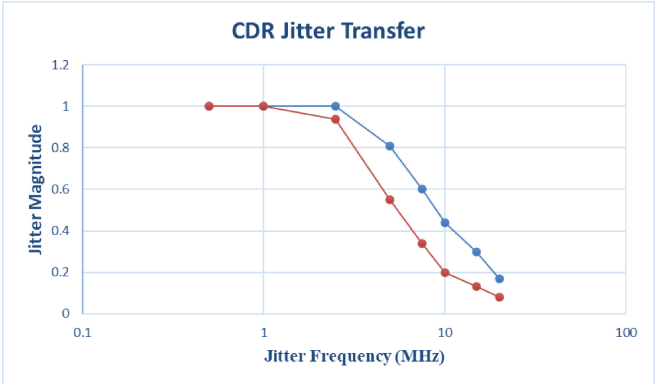

FIGURE 28. CDR loop Jitter Transfer curves for operation with 6Bit (Blue line) and 8-Bit (Red-line) loop filter respectively.

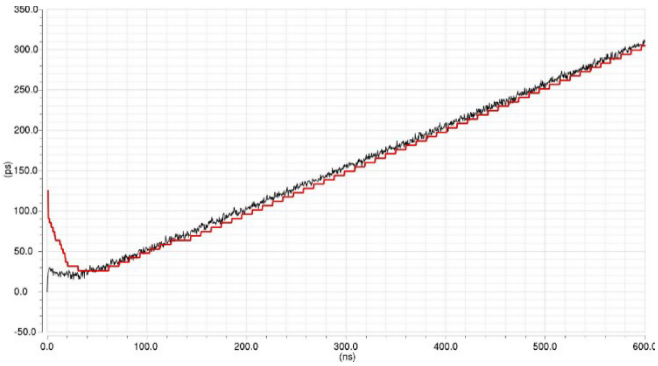

FIGURE 29. CDR clock phase (red line) versus Data sequence phase (black line), for operation with 8-Bit loop filter under 500ppm frequency offset.

a constantly decreased portion of input jitter is transferred to the output of the CDR as F_{SJ} is increased.

The results derived from the Jitter Transfer simulation for operation with 6-Bit and 8-Bit loop filters are summarized in Table 3 while the resulting Jitter transfer curves for operation at 5.83 Gbps are illustrated in Fig. 28. The Jitter transfer curves represent the closed loop transfer function of the system. From the results, it is confirmed that the CDR loop exhibits a low pass characteristic with a cut-off frequency near 6 MHz and 4 MHz in case of using a 6-Bit or a 8-Bit Loop filter respectively.

Subsequently in Fig. 29 is presented the CDR clock phase (Red line) versus the input Data sequence phase (black line)

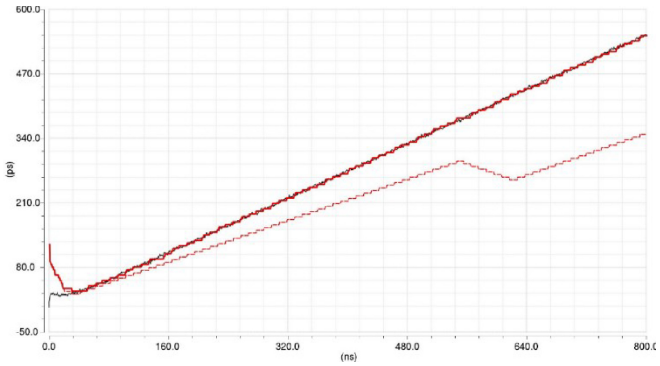

FIGURE 30. CDR clock phase (red line) versus Data sequence phase (black line), for operation with 8-Bit and 6-Bit loop filter under 800ppm frequency offset.

TABLE 4. CDR loop performance summary.

Performance Parameter	*DIG-LF*	
	6-Bit	*8-Bit*
Technology	TSMC 65nm	
Supply	1.0V	
Data rate	5.83Gbps	5.83Gbps
Jitter Generation	5.4ps	5.4ps
Jitter tolerance @1MHz)	$1.7UI_{pp}$	$1UI_{pp}$
BER	$>10^{-10}$	$>10^{-10}$
Power Consumption	13.5mW	14.2mW
Max.freq offset	>800ppm	500ppm

TABLE 5. Comparison.

Design	*[24]*	*[25]*	*[36]*	*[37]*	*This work*
CMOS Proccess (nm)	90	65	40	90	65
Data rate (Gbps)	8-28	1-16	1.29-5.184	20	5.83
Power per (mW/Gbps)	3.9	5.47	2.4	5.1	2.3
Max freq. offset (ppm)	122	-	-	-	>500
Locking time (UI)	-	-	20	1	>30
Meas. /Simul.	Meas	Meas		Meas	Simul

as a function of time, for operation at 5.83 Gpbs with 8-Bit loop Filter. In this case in order to test the CDR slew rate, a constant frequency offset has been inserted between the clock and Data signals equal to 1.5 MHz (i.e., 500 ppm), which is translated in a slightly different rate between the two signals. More specifically the data signal is rated at 5.833 Gbps while the CDR clock signal at 5.830 Gbps. As is shown the loop is able to properly track this frequency offset and remain locked. In Fig. 30 the same metrics are presented, but in this case for operation with a 6-Bit CDR loop filter and a frequency offset between the two signals equal to 2 MHz (i.e., 800 ppm). In the same diagram with red dashed line is presented the CDR clock phase for operation under the same frequency offset but with 8-Bit loop filter.

As was shown for operation with a 6-Bit loop filter the CDR is able to tolerate the frequency offset and remain locked. Instead with the 8-Bit loop filter, the loop is not able to track properly the frequency offset and eventually loses the phase locking and leads the operation to fail. According to simulation results the maximum frequency offset that the CDR is able to tolerate for operation with 6-Bit and 8-Bit loop filters, is 800 ppm and 500 ppm respectively. This result was expected owing to the fact that a lower loop bandwidth degrades the slew rate of the loop as explained in Section III-D. Finally, the most important performance characteristics of the CDR loop are summarized in Table 4.

Comparison with the measurement results of some of the state-of-the-art works is presented in Table 5. Although this work is based on simulations it looks competitive in terms of power consumption per data rate and presents good minimum frequency offset tracking ability. However, the most important key point of this work is to present the basic steps in details as a starting point for designing and jitter optimized PI-based CDR by means of jitter models for pre-fabrication verification.

VII. CONCLUSION

In this paper, an optimized, in terms of jitter impact, phase-interpolator-based clock and data recovery circuit has been presented. First, a brief overview of the most commonly used circuitry architectures in high-speed serial links for clock and data recovery is given. Then, their main performance characteristics and design challenges are presented. According to the proposed jitter analysis and the proposed methodology a CMOS 8-Bit dual-loop CDR is proposed, designed with the TSMC 65 nm process node. The CDR is based on a proposed PI topology with extended phase resolution. The developed, in this work, CDR loop has a minimum frequency offset tracking ability equal to 500 ppm at 5.83 Gbps, and so is suitable for adoption either in mesochronous or plesiochronous HSSI receivers. It consumes 14.2 mW under 1.0 V supply voltage and is able to achieve better than 10^{-10} BER performance. The CDR loop performance verification has been realized through the AMS simulator of Analog Design environment of Cadence, by co-simulations of the transistor level CDR circuit with a Verilog-AMS based jitter generator.

REFERENCES

[1] B. Casper, "Clocking wireline systems: An overview of wireline design techniques," *IEEE Solid-State Circuits Mag.*, vol. 7, no. 4, pp. 32–41, 2015.
[2] B. Razavi, *Design of Integrated Circuits for Optical Communications*. Hoboken, NJ, USA: Wiley, 2012.
[3] H. C. Lee, *An Estimation Approach to Clock and Data Recover*, Stanford Univ., Stanford, CA, USA, 2006.
[4] M.-T. Hsieh and G. Sobelman, "Architectures for multi-gigabit wire-linked clock and data recovery," *IEEE Circuits Syst. Mag.*, vol. 8, no. 4, pp. 45–57, 1st Quart., 2008.
[5] D. K. Weilander, *Precision CMOS Receivers for VLSI Testing Applications*, Stanford Univ., Stanford, CA, USA, 2001.
[6] J. Kim and D.-K. Jeong, "Multi-gigabit-rate clock and data recovery based on blind oversampling," *IEEE Commun. Mag.*, vol. 41, no. 12, pp. 68–74, Dec. 2003.
[7] S. I. Ahmed and T. A. Kwasniewski, "Overview of oversampling clock and data recovery circuits," in *Proc. Can. Conf. Electr. Comput. Eng.*, Saskatoon, SK, Canada, 2005, pp. 1876–1881.
[8] M. Nogawa et al., "A 10 Gb/s burst-mode CDR IC in 0.13/spl mu/m CMOS," in *IEEE Int. Dig. Tech. Papers Solid-State Circuits Conf. (ISSCC)*, San Francisco, CA, USA, 2005, pp. 228–595.

[9] A. Tajalli, P. Muller, M. Atarodi, and Y. Leblebici, "A multichannel 3.5mW/Gbps/channel gated oscillator based CDR in a 0.18μm digital CMOS technology," in *Proc. 31st Eur. Solid-State Circuits Conf. (ESSCIRC)*, Grenoble, France, 2005, pp. 193–196.

[10] A. Tajalli, P. Muller, and Y. Leblebici, "A power-efficient clock and data recovery circuit in 0.18μm CMOS technology for multi-channel short-haul optical data communication," *IEEE J. Solid-State Circuits*, vol. 42, no. 10, pp. 2235–2244, Oct. 2007.

[11] X. Zhao, Y. Chen, X. Zheng, P. Mak, and R. P. Martin, "A 0.01-mm2 1.2-pJ/bit 6.4-to-8Gb/s reference-less FD-less BBCDR using a deliberately-clock-selected strobe point based on a 2π/3-interval phase," in *IEEE MTT-S Int. Microw. Symp. Tech. Dig.*, Atlanta, GA, USA, 2021, pp. 386–389, doi: 10.1109/IMS19712.2021.9574983.

[12] F.-T. Chen, J.-M. Wu, J. Y.-C. Liu, and M.-C. F. Chan, "A 100Gb/s quad-rate transformer-coupled injection-locking CDR circuit in 65nm CMOS," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Beijing, China, 2013, pp. 1950–1953, doi: 10.1109/ISCAS.2013.6572250.

[13] A. Tsimpos, A. Demartinos, S. Vlassis, and G. Souliotis, "Jitter tolerance modeling and calibration for high-speed serial interfaces," *Integration*, vol. 57, pp. 101–107, Mar. 2017.

[14] S. Sidiropoulos and M. A. Horowit, "A semidigital dual delay-locked loop," *IEEE J. Solid-State Circuits*, vol. 32, no. 11, pp. 1683–1692, Nov. 1997.

[15] S. Palermo, *Design of High-Speed Optical Interconnect Tranceivers*, Stanford Univ., Stanford, CA, USA, 2007.

[16] G. Souliotis, C. Laoudias, F. Plessas, and T. Nikolaos, "Phase interpolator with improved linearity," *Circuits, Syst., Signal Process.*, vol. 35, no. 2, pp. 367–383, Feb. 2016.

[17] A. J. Deka and V. Prasann, "A 1Gbps—10 Gbps multi-standard auto-calibrated all digital phase interpolator in 14nm CMOS," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Lisbon, Portugal, 2015, pp. 2209–2212, doi: 10.1109/ISCAS.2015.7169120.

[18] B. Razavi, *RF Microelectronics*, 2nd ed. Los Angeles, CA, USA: Prentice Hall, 2012.

[19] S. Palermo, "High-speed links circuits and systems," ECEN 720 Cource Notes, Texas A&M Univ., College Station, TX, USA, 2019.

[20] E. Evans, *ADN2817/ADN2818 BER Monitor and Sample Phase Adjust User Guide*, Analog Devices, Wilmington, MA, USA, 2015.

[21] S. Palermo, "High-speed serial I/O design for channel-limited and power-constrained systems," in *CMOS Nanoelectronics Analog and RF VLSI Circuits*. New York, NY, USA: McGraw-Hill, 2011.

[22] Y. Liu, Y. Wang, S. Jia, G. Zhang, and X. Zhang, "A 6.25Gb/s CMOS clock and data recovery DLL with anti-harmonic lock," in *Proc. IEEE Int. Conf. Electron Devices Solid-State Circuits*, Chengdu, China, 2014, pp. 1–2, doi: 10.1109/EDSSC.2014.7061227.

[23] N. Kalentari and J. F. Buckwalter, "A multichannel serial link receiver with dual-loop clock-and-data recovery and channel equalization," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 11, pp. 2920–2931, Nov. 2013.

[24] C. Kromer, G. Sialm, C. Menolfi, M. Schmatz, F. Ellinger, and H. Jackel, "A 25-Gb/s CDR in 90-nm CMOS for high-density interconnects," *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2921–2929, Dec. 2006.

[25] G. Wu et al., "A 1–16 Gb/s all-digital clock and data recovery with a wideband high-linearity phase interpolator," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 24, no. 7, pp. 2511–2520, Jul. 2016.

[26] A. Tsimpos, A. C. Demartinos, G. Souliotis, and S. Vlassis, "Multi-rate phase interpolator for high speed serial interfaces," *Microelectron. J.*, vol. 54, pp. 40–47, Aug. 2016, doi: 10.1016/j.mejo.2016.05.008.

[27] A. Demartinos, A. Tsimpos, S. Vlassis, S. Sgourenas, and G. Souliotis, "A 3GHz VCO suitable for MIPI M-PHY serial interface," in *Proc. 10th Int. Conf. Design Technol. Integr. Syst. Nanoscale Era (DTIS)*, Napoli, Italy, 2015, pp. 1–6.

[28] A. Demartinos, A. Tsimpos, V. Spyridon, G. Souliotis, and S. Sgourenas, "A scalable voltage controlled oscillator for multi-rate high-speed interfaces," *Microelectron. J.*, vol. 55, no. 9, pp. 134–142, 2016.

[29] J. D. H. Alexander, "Clock recovery from random binary data," *Electron. Lett.*, vol. 11, pp. 541–542, Oct. 1975.

[30] J. Lee, K. S. Kunbert, and B. Razavi, "Analysis and modeling of bang-bang clock and data recovery circuits," *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1571–1580, Sep. 2004.

[31] K. Kundert, *Verification of Bit-Error Rate in Bang-Bang Clock and Data Recovery Circuits*, The Designers Guide Community, Los Altos, CA, USA, 2010.

[32] M. R. Müller, "Digital centric multi-gigabit SerDes design and verification," Ph.D. dissertation, Faculty Nat. Sci. Math., Heidelberg Univ., Heidelberg, Germany, 2018.

[33] G. Lee, Y. Sub Kim, J. H. Lee, D. H. Choi, and S. Kim, "A 1.0 Gb/s BiCMOS multi-channel optical interface transmitter and receiver chip set for high resolution digital displays," *IEEE Trans. Consum. Electron.*, vol. 47, no. 3, pp. 273–277, Aug. 2001.

[34] T.-C. Hsueh et al., "26.4 A 25.6Gb/s differential and DDR4/GDDR5 dual-mode transmitter with digital clock calibration in 22nm CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, San Francisco, CA, USA, 2014, pp. 444–445.

[35] T. Tikka, K. Stadius, J. Ryynanen, and M. Voutilainen, "A 1.2—6.4 GHz clock generator with a low-power DCO and programmable multiplier in 40-nm CMOS," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Melbourne, VIC, Australia, 2014, pp. 506–509.

[36] K. Maruko et al., "A 1.296-to-5.184Gb/s transceiver with 2.4mW/(Gb/s) burst-mode CDR using dual-edge injection-locked oscillator," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, San Francisco, CA, USA, 2010, pp. 364–365.

[37] J. Lee and M. Liu, "A 20Gb/s burst-mode CDR circuit using injection-locking technique," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, San Francisco, CA, USA, 2007, pp. 46–586.

GEORGE SOULIOTIS (Senior Member, IEEE) received the B.Sc. degree in physics from the University of Ioannina, Greece, in 1993, and the M.Sc. and Ph.D. degrees in electronics from the University of Patras, Greece, in 1998 and 2003, respectively. He served in several academic positions and he is currently an Assistant Professor with the Department of Electrical and Computer Engineering, University of Peloponnese, Patras. He is the author of more than 50 papers in high level journals, conferences and of two patents. He has more than 20 years professional and research experience in the design of analog integrated circuits and systems. He is a reviewer in more than ten international journals.

ANDREAS TSIMPOS (Member, IEEE) received the B.Sc. degree in physics and the M.Sc. and Ph.D. degrees in electronics from the University of Patras, Greece, in 2010, 2013, and 2020, respectively. He has served as a Teaching Assistant with Electronics Laboratory, Department of Physics from 2011 to 2017. He is currently with Adveos Microelectronics, Athens, Greece, working in the design of RF circuits for wireless and wireline applications. His research focuses on clocking and synchronization circuits and systems (PLLs/DLLs/CDRs) for SerDes and RF transceivers. He is a member of the IEEE Solid-State Circuits Society and the IEEE Circuits and Systems Society.

SPYRIDON VLASSIS received the B.Sc. degree in physics, the M.Sc. degree in electronic physics, and the Ph.D. degree from the Aristotle University of Thessaloniki, Greece, in 1994, 1996, and 2000, respectively. He was working as a Senior Engineer for VC funded startup companies in the development and commercialization of high-performance RFICs for wireless communications and power management ASIC for consumer applications. He is a Full Professor with the Electronics Laboratory, Department of Physics, University of Patras, Greece. He holds six patents. His research interests are in analog and mixed-signal integrated circuits and signal processing.