

# A 22.5-to-32-Gb/s 3.2-pJ/b Referenceless Baud-Rate Digital CDR With DFE and CTLE in 28-nm CMOS

Wahid Rahman, Danny Yoo, Joshua Liang, Ali Sheikholeslami, *Senior Member, IEEE*,  
Hirotaka Tamura, *Fellow, IEEE*, Takayuki Shibasaki, and Hisakatsu Yamaguchi

**Abstract**—This paper presents a referenceless baud-rate clock and data recovery (CDR) incorporated with a continuous-time linear equalizer (CTLE) and one-tap decision feedback equalizer (DFE) to achieve data rates from 22.5 to 32 Gb/s across a channel with Nyquist loss ranging from 10.1 to 14.8 dB. The referenceless CDR includes a proposed frequency acquisition scheme that consists of two parts: frequency detection and frequency correction. Frequency detection is achieved by examining rising and falling data waveforms to detect discrepancies between the data rate and the locally recovered clock frequency. Frequency correction uses digitally adjustable asymmetry of the proposed adjustable baud-rate phase detector to correct any frequency error. The receiver is implemented in the TSMC 28-nm CMOS process with an analog front end consisting of a CTLE, sampling comparators, a digitally controlled oscillator, and a digital back end consisting of synthesized digital CDR logic. The open-loop frequency detector range is measured to be 39%. The closed-loop CDR capture range is measured to be 34%, limited by test equipment. The proposed frequency acquisition scheme improves the measured CDR capture range by up to 227×. At 32 Gb/s, the entire receiver consumes 102.04 mW, achieving energy consumption below 3.19 pJ/b.

**Index Terms**—Baud rate, clock and data recovery (CDR), continuous-time linear equalizer (CTLE), decision feedback equalizer (DFE), digital CDR, frequency detection, referenceless CDR.

## I. INTRODUCTION

AS DATA rates in high-speed wireline links continue to increase, baud-rate clock and data recovery (CDR) circuits serve as an alternative to conventional clock recovery techniques. Conventional oversampling designs, such as 2× oversampling CDRs, sample the incoming data waveform more than once per unit interval (UI) to recover phase error information [1]–[6]. As shown in Fig. 1(a), samples on the data clock phase ( $D_n$ ) and samples on the edge clock phase ( $E_n$ )

Manuscript received April 24, 2017; revised July 8, 2017 and August 9, 2017; accepted August 14, 2017. Date of publication October 16, 2017; date of current version November 21, 2017. This paper was approved by Guest Editor Mounir Meghelli. (Corresponding author: Wahid Rahman.)

W. Rahman was with the University of Toronto, Toronto, ON M5S 3G4, Canada. He is now with AlphaWave IP, Toronto, ON M1B 5H6, Canada (e-mail: wahid.rahman@alum.utoronto.ca).

D. Yoo and A. Sheikholeslami are with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada.

J. Liang was with the University of Toronto, Toronto, ON M5S 3G4, Canada. He is now with Huawei, Toronto, ON L3R 5A4, Canada.

H. Tamura, T. Shibasaki, and H. Yamaguchi are with Fujitsu Laboratories Ltd., Kawasaki 211-8588, Japan.

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/JSSC.2017.2744661

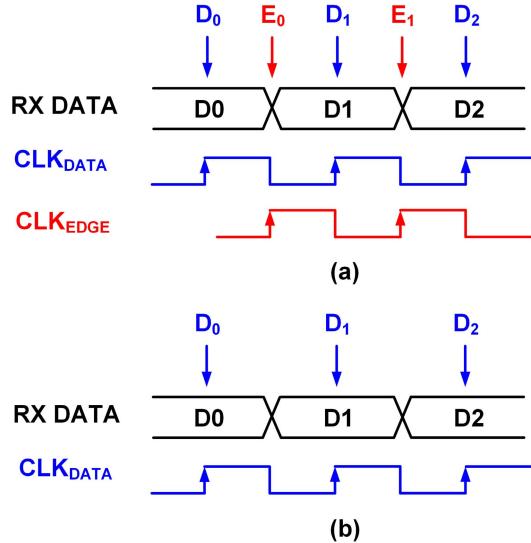


Fig. 1. Comparison of (a) 2× oversampling and (b) baud-rate CDRs.

are retrieved to achieve 2× oversampling. In typical 2× oversampling architectures, the edge clock samples are required in conjunction with the data clock samples for the CDR's phase detector (PD) to determine timing recovery decisions. On the other hand, baud-rate clocking eliminates the need for additional samples: only the  $D_n$  samples are required [7]–[13], as shown in Fig. 1(b). This allows for a reduced number of high-speed front-end samplers required in the receiver compared with conventional oversampling CDRs. As well, fewer high-speed clocks need to be routed through the clock distribution network, allowing for more energy-efficient receivers.

In addition to their sampling architecture, CDRs can also be classified as referenced or referenceless. Referenced CDRs require a reference clock to recover data. This reference clock, or its filtered version generated by a phase-locked loop, maintains a frequency close to the expected incoming data rate. The CDR then uses this reference in conjunction with a phase interpolator [8], [9], [11], [12], a delay-locked loop [6], or a blind clocking scheme [4], [10] to recover the data. CDRs which do not use a reference clock but instead require external manual tuning of an on-chip local oscillator are also considered as referenced CDRs. The role of these manually tuned signals is to bring the initial voltage-controlled oscillator (VCO) frequency to within the CDR's capture range before starting the recovery operation [5], [13], and thus such signals are still considered as an external reference to the CDR.

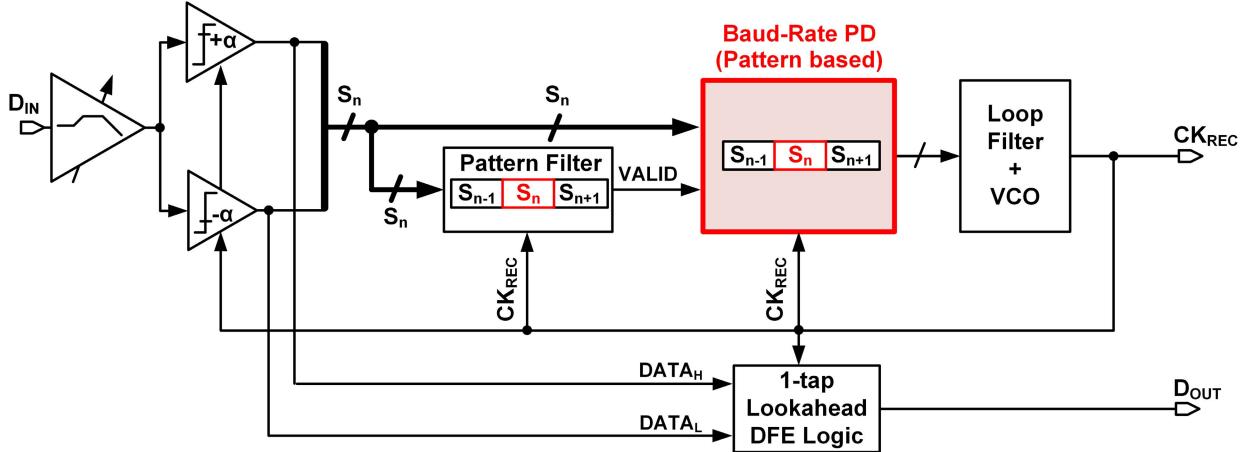


Fig. 2. Baud-rate referenced CDR architecture from [13].

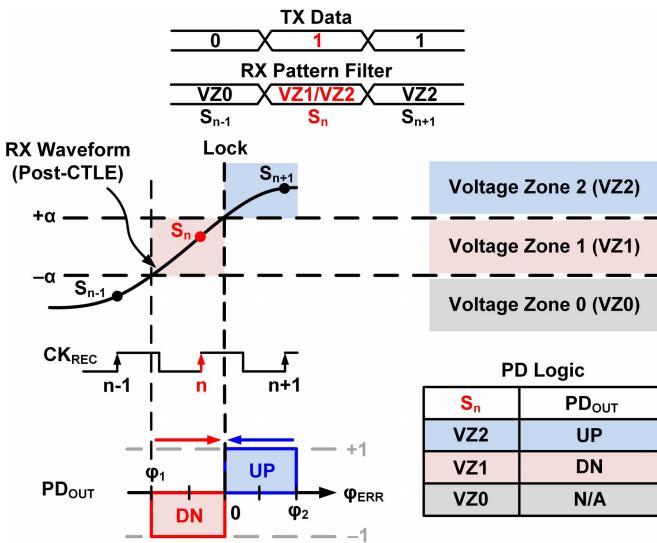


Fig. 3. Operation of baud-rate PD from [13].

In contrast, referenceless CDRs do not require any external reference signals. Instead, they recover the required clock relying solely on the incoming data waveform. Referenceless CDRs must automatically adjust the frequency of their recovered clock to match the incoming data rate. To do so, additional frequency detection and correction schemes are required. The main operating principle of such schemes is to observe and correct the frequency error between the incoming data and the recovered clock. Detecting frequency error typically involves observing how phase error changes due to frequency offset [2], [3], [14]–[16]. Correcting frequency error can be accomplished using a dedicated control loop in the CDR in addition to a traditional phase loop [2], [15]. Other correction mechanisms utilize an automatic phase adjustment scheme to correct frequency error without an additional loop [3], [14], [16].

Regardless of frequency detection and correction mechanisms, referenceless CDRs eliminate the need for external crystal oscillators or manual tuning mechanisms. Such CDRs are attractive for retimer ICs in optical or backplane applications where including external crystal oscillators are costly

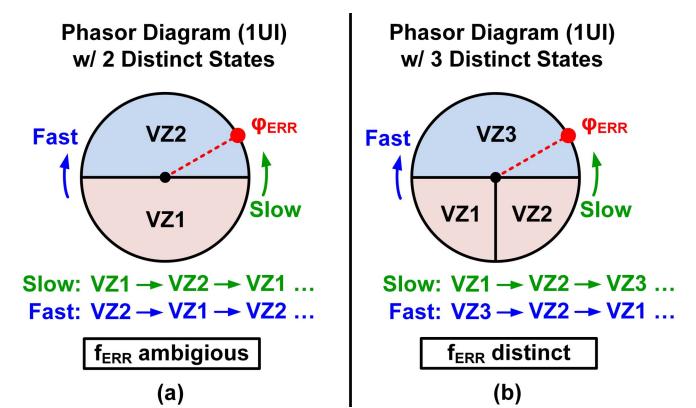


Fig. 4. Attempting to detect frequency error using a phasor diagram with (a) two distinct detection states and (b) three distinct detection states.

and pins are limited. In addition, referenceless CDRs achieve a significantly extended capture range compared with their referenced counterparts, and thus can automatically adapt to multiple data rates if required.

The current research challenge remains in achieving a referenceless CDR that operates at baud-rate sampling. The push for baud-rate sampling becomes more evident as energy-efficient, high data-rate receivers are required. Moreover, at higher data rates, Nyquist loss is non-negligible. Existing referenceless designs rely on oversampled CDRs and data inputs with relatively sharp transitions coming from a low-loss channel; they do not extend to baud-rate sampling or to high data rates where channel loss is significant.

To address this issue, this paper presents the first baud-rate referenceless CDR. In particular, this paper presents further details of the work presented in [17]. We present a frequency detection and correction scheme that enables an existing baud-rate sampling scheme [13] to operate as a referenceless baud-rate CDR. The proposed referenceless design operates using only one data sample per UI and does not require sharp data transitions: the design tolerates up to 14.8 dB Nyquist channel loss. The correction scheme requires a single CDR control loop, eliminating the need for two separate CDR control loops. The scheme extends the capture range by up to

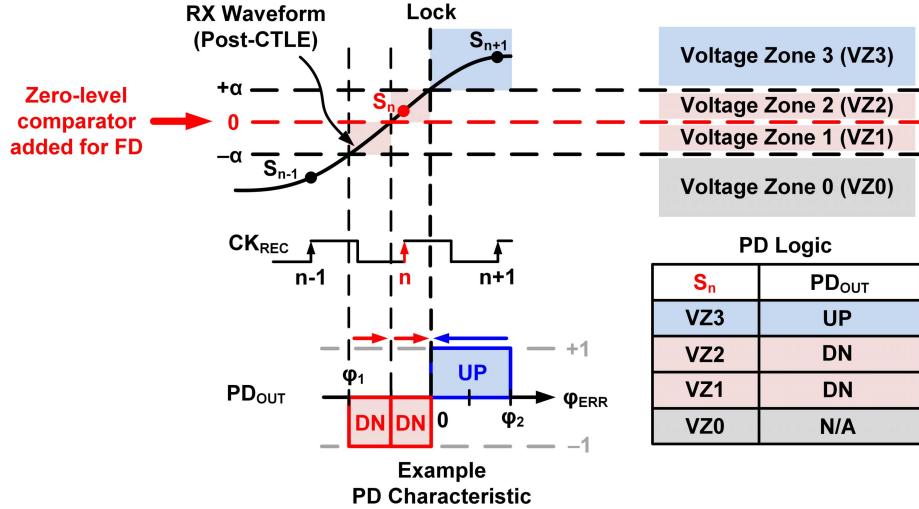


Fig. 5. Additional comparator added for purposes of frequency detection.

227 $\times$  compared with the referenced CDR while maintaining competitive energy efficiency.

The remainder of this paper is organized as follows. Section II reviews current referenceless CDRs as well as baud-rate referenced CDRs, and their respective limitations. Section III presents the proposed design and Section IV presents the implementation details. Section V presents the measurement results and Section VI concludes this paper.

## II. BACKGROUND

### A. Referenceless CDRs

State-of-the-art referenceless CDRs implement a frequency acquisition scheme to correct the locally recovered clock frequency to the incoming data rate. Several proposed methods exist to accomplish this [14]–[16], [18], [19]. Frequency acquisition typically involves two components, frequency detection and frequency correction, and existing works can be categorized by their approach to each. For example, one approach to detect frequency is utilizing acquisition logic to search for the matching clock frequency based on PD outputs [15], [18]. In [15], the local oscillator must be reset to its lowest frequency before initializing its proposed frequency search, and thus is limited to unidirectional frequency correction. The drawback with this approach is prolonged acquisition time, although techniques such as gear-shifting can improve this. In [18], the frequency error may initially increase before its search can detect it, which again can prolong lock time. This frequency detection by searching is avoided in [14] and [16]. However, these works rely on the data edge to sample high-speed analog clocks; a sharp data transition is required for this sampling, which may not be available in channels with significant channel loss without power-hungry equalization. Frequency detectors (FDs) based on classical rotational or digital quadri-correlator frequency detection have also been presented [19], but they suffer from the same data sampling clock issue. As well, they often require additional clock or data phases and are therefore not suited for true baud-rate operation.

Once frequency error is detected, frequency correction schemes must be considered. The work in [15] relies on a dual-loop CDR architecture, where a frequency-locked loop controls the local oscillator in parallel with the CDR's conventional phase-locked loop. Careful design considerations and control hardware are required for such dual-loop CDRs to ensure that the two loops do not interfere or conflict in their concurrent control of the local oscillator during phase lock. To avoid this issue, the works in [14], [16], and [18] propose tuning the PD characteristic using tunable delay elements, such that the phase-locked loop naturally corrects the frequency error. However, such delay elements are placed on the high-speed clock or data paths and can limit performance for faster data rates.

Most crucially, the state-of-the-art referenceless CDRs do not support baud-rate CDRs: [14] and [18] operate using linear, analog PDs, while [15] and [16] rely on oversampled bang-bang PDs. Thus, in addition to addressing the drawbacks mentioned earlier, there exists an even further design gap for a referenceless CDR to address these challenges while operating at baud rate.

### B. Baud-Rate PD

With the push to higher data rates, several recent works have proposed baud-rate referenced CDRs. Building on the original Mueller–Müller baud-rate sampling [7], these recent works attempt to further improve the energy efficiency in clock recovery [8]–[13]. Fig. 2 presents a high-level block diagram of the most recent referenced baud-rate CDR [13]. The CDR incorporates a one-tap look-ahead decision feedback equalizer (DFE) with a pattern-based baud-rate PD. The PD and DFE share two front-end comparators, which sample the incoming data partially equalized by a continuous-time linear equalizer (CTLE). The comparators are set to the first post-cursor ISI values  $\pm\alpha$  for correct DFE operation. The output of these comparators generates a sample,  $S_n$ . A sequence of three samples are used by a pattern filter and the pattern-based baud-rate PD. If a valid pattern is detected, the PD uses the

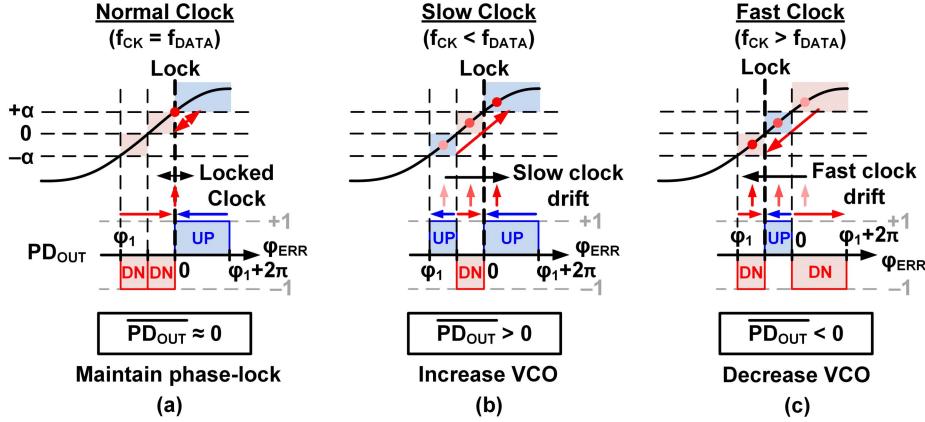


Fig. 6. Adjustable baud-rate PD for when the recovered clock (a) matches the data rate, (b) is slower than the data rate, and (c) is faster than the data rate.

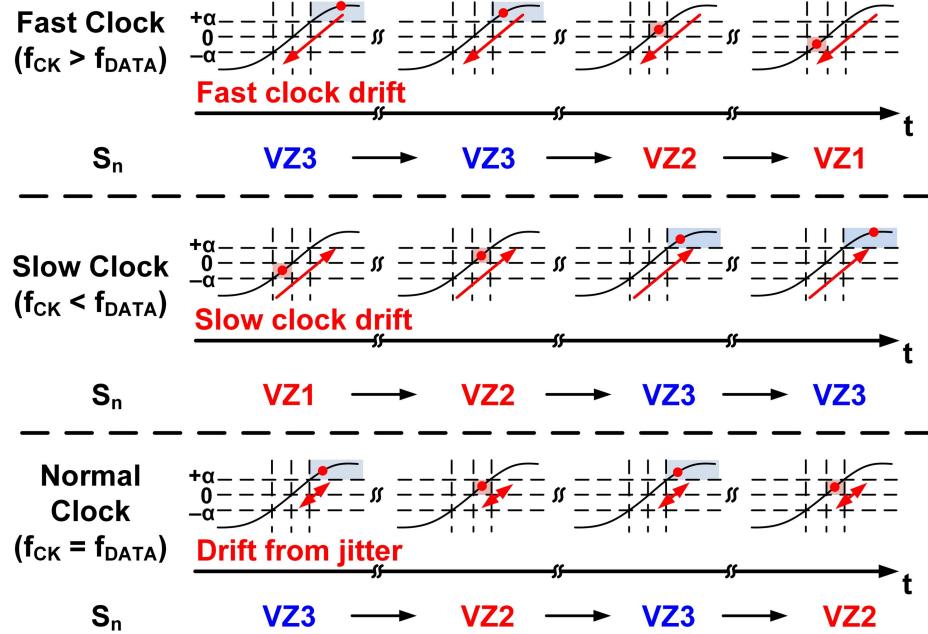


Fig. 7. Detecting frequency error using three detection zones.

middle sample of the pattern to determine its early or late decision, which then controls the loop filter and local oscillator.

Fig. 3 shows the details of the pattern-based baud-rate PD. The PD relies on a rising or falling waveform (falling not shown in the figure) for being sampled by the front-end comparators. The front-end comparators divide the voltage scale into three zones, which from henceforth will be referred to as “voltage zones” VZ0, VZ1, and VZ2. The pattern filter extracts rising waveforms that are deemed valid for timing recovery:  $S_{n-1}$  is required to be a definite “0” bit (VZ0), and  $S_{n+1}$  is required to be a definite “1” bit (VZ2). The middle sample  $S_n$  is required to be a “1” bit as well, but due to channel ISI, it is degraded and can lie in either zones VZ1 or VZ2. The work in [13] defines the PD lock position as the crossing between the rising waveform and the  $+\alpha$  threshold, and whether  $S_n$  lies in VZ1 or VZ2 defines the PD early or late decisions, respectively. Note that VZ0 is only used for pattern filtering purposes, and cannot be used for a PD detection state.

Only two detection states, zones VZ1 and VZ2, are available and sufficient for a bang-bang PD.

The ability of this scheme to operate as a frequency detection scheme is limited. This can be illustrated in the phasor diagram of Fig. 4. The phasor diagram represents 1 UI of phase error  $\phi_{\text{ERR}}$  ( $= \phi_{\text{DATA}} - \phi_{\text{CLK}}$ ) around the  $360^\circ$  of a circle. In the presence of frequency error  $f_{\text{ERR}}$  ( $= f_{\text{DATA}} - f_{\text{CLK}}$ ), the point representing  $\phi_{\text{ERR}}$  rotates around the circle counterclockwise for positive (slow) frequency error and clockwise for negative (fast) frequency error. Fig. 4(a) represents how the two detectable voltage zones VZ1 and VZ2 map to the phasor diagram. While sufficient for bang-bang PD operation, having only two detectable zones is not sufficient to determine the direction of frequency error.

### III. PROPOSED BAUD-RATE FREQUENCY DETECTOR

#### A. Proposed Detection Zones

To address the issue of frequency detection, a third zone to detect phase error is proposed. As shown in Fig. 4(b),

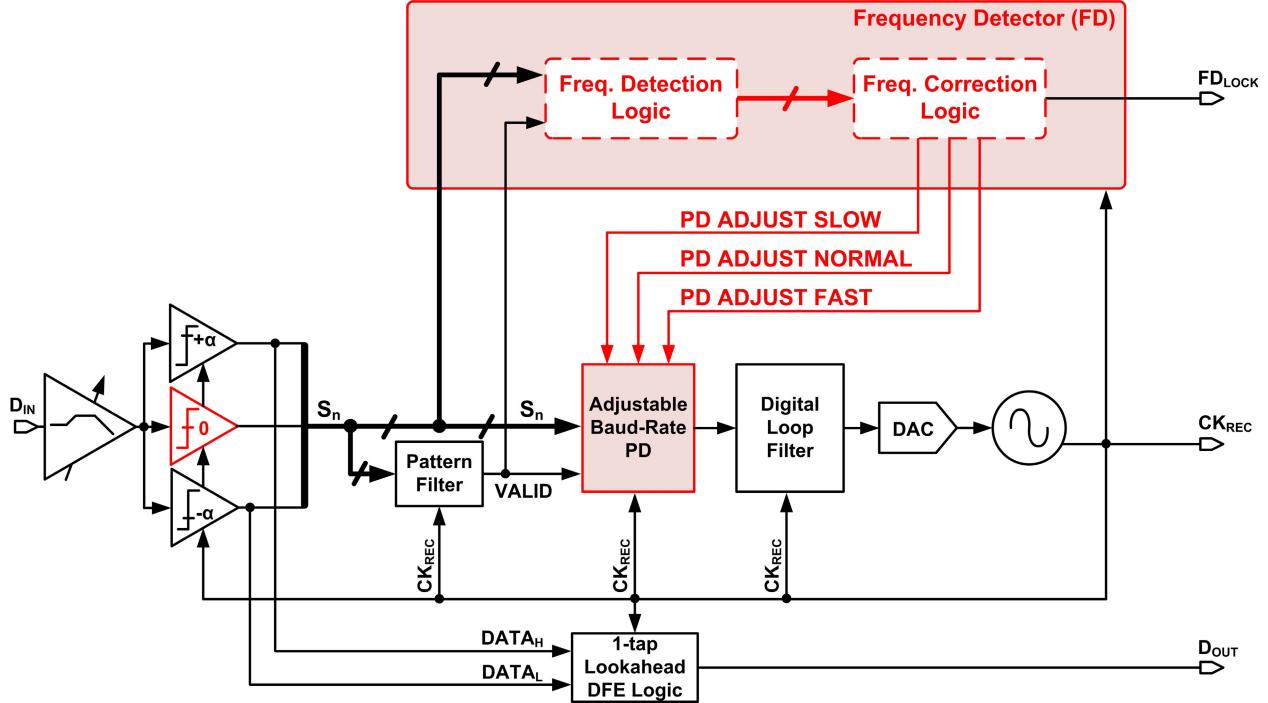


Fig. 8. Proposed CDR architecture. Components critical to frequency acquisition are highlighted in red.

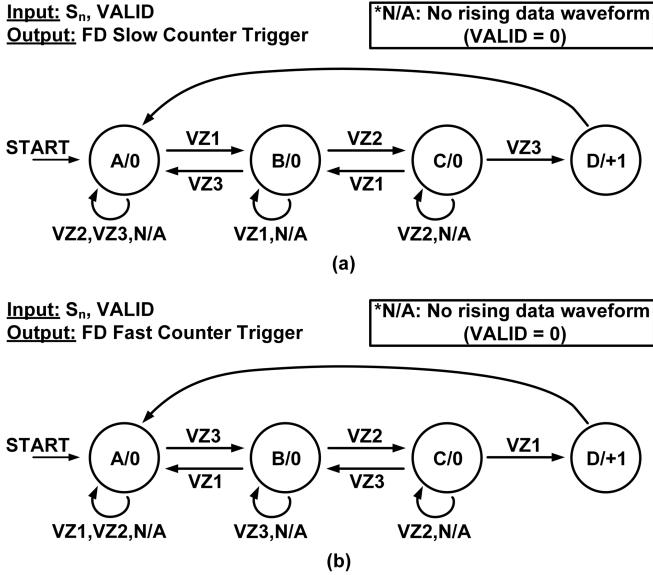


Fig. 9. State machines to detect (a) 1-2-3 transitions for slow clocks and (b) 3-2-1 transitions for fast clocks.

this is achieved by dividing an existing zone into two. Doing so then allows distinguishing positive (slow) and negative (fast) frequency errors. Note that these zones refer to the unique detection states available for tracking the phase error, not the total number of zones used to quantize the data waveform.

The proposed method for detecting the third zone is by adding an additional comparator level to the existing baud-rate CDR architecture [13]. This comparator level, as shown in Fig. 5, is set at the zero-level, halfway between the

$+\alpha$  and  $-\alpha$  comparator levels. Using a similar pattern filtering to extract rising waveforms, the filtered  $S_n$  can be resolved to three unique mappings, Zones 1, 2, and 3. Having three unique detection zones allows the sign of the frequency error to be determined.

The three unique detection zones also assist in frequency error correction. Fig. 5 shows the mapping from filtered  $S_n$  voltage zones to PD output values. This mapping defines the PD logic, and results in the corresponding PD characteristic. Crucially, this mapping does not need to be static. If this mapping is dynamic, and it is implemented using synthesized digital logic, then the PD characteristic can be changed as needed to assist with frequency acquisition.

This is the central thesis to the proposed frequency correction scheme: an adjustable baud-rate PD. The PD output is made asymmetric, and this asymmetry produces the desired positive or negative PD output to correct frequency error. As shown in Fig. 6, the PD characteristic can be adjusted based on three cases of frequency error: the recovered clock matches the data rate (Fig. 6(a)), the recovered clock is slower than the data rate (Fig. 6(b)), and the recovered clock is faster than the data rate (Fig. 6(c)). By adjusting the PD characteristic, the average PD output can be adjusted to be positive or negative. As phase error changes over time due to any existing frequency error, the average PD output then increases or decreases the CDR's local oscillator frequency (e.g. a digitally controlled oscillator (DCO)) to correct the frequency error. Note that in all three cases, a stable lock point is maintained in the PD characteristic. Doing so allows the CDR to automatically lock once frequency error is corrected: no additional mechanism is required to monitor

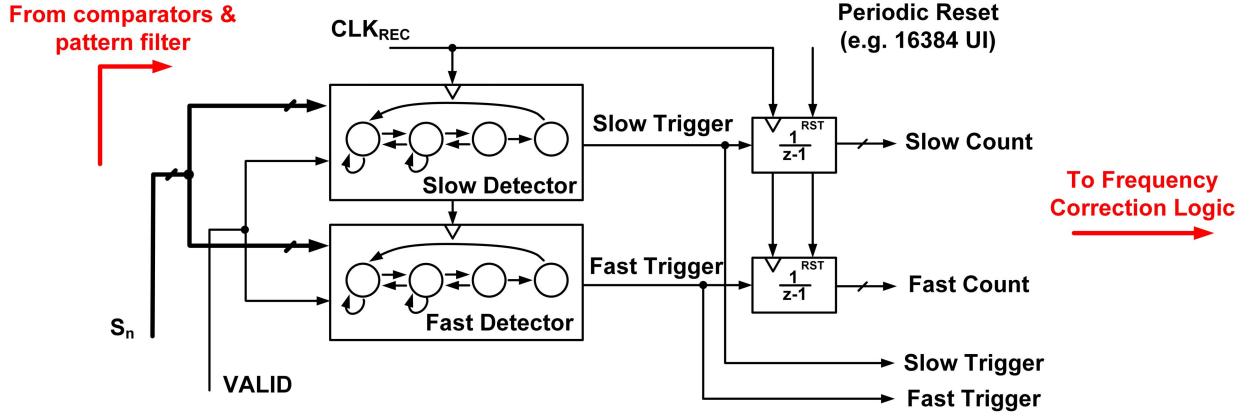


Fig. 10. Frequency detection logic.

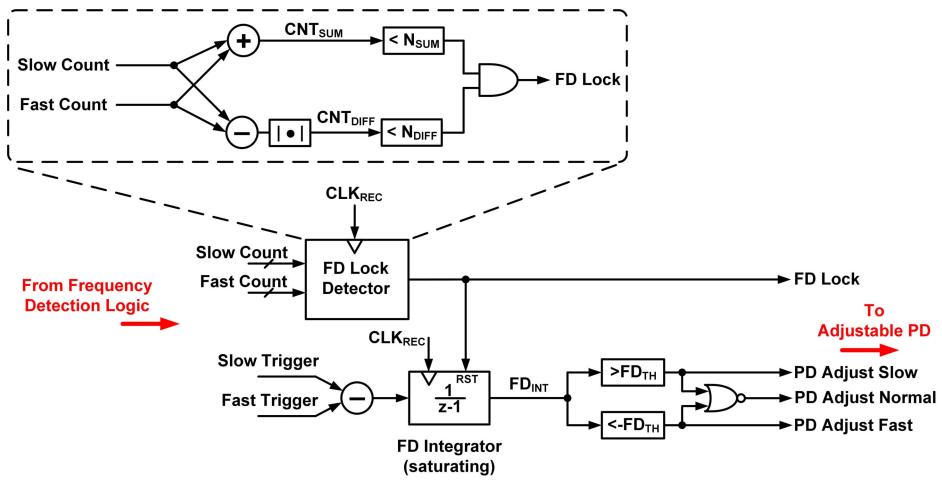


Fig. 11. Frequency correction logic.

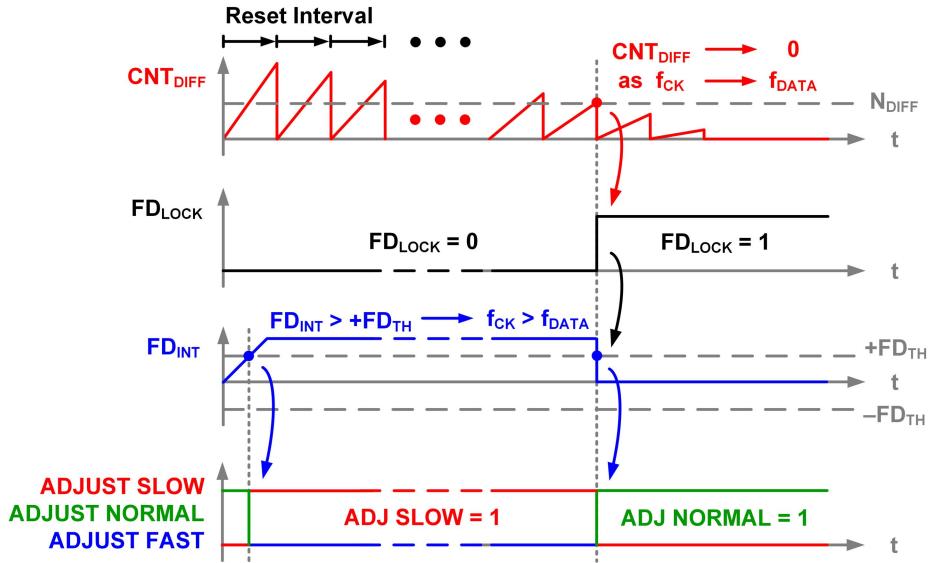


Fig. 12. Conceptual illustration of frequency correction transient behavior.

overcorrection. For example, if the clock is too fast, the PD characteristic is adjusted during frequency acquisition to produce a negative average PD output (Fig. 6(c)). This reduces

the clock frequency until the frequency error is corrected. Due to the modified PD response, the CDR lock point is shifted from its nominal phase during frequency acquisition. However,

once the CDR verifies that frequency error is corrected, the PD characteristic is returned to normal (Fig. 6(a)). The CDR must then adjust the clock phase to target the true, desired lock point defined by the normal PD characteristic. In this way, the CDR resumes correcting for phase error.

Detecting frequency error using the three distinct zones is shown in Fig. 7. The principle of operation relies on monitoring how the filtered sample  $S_n$  “drifts” along the rising waveform over time due to frequency error. If the recovered clock is too fast, the sample  $S_n$  drifts down the waveform over time, transitioning from zones 3 to 2 to 1. Such a 3-2-1 zone transition indicates to the FD of a possible fast clock instance. Similarly, for a slow clock, the sample  $S_n$  drifts up the rising waveform over time; 1-2-3 zone transitions indicate a possible slow clock instance. The reason why such transitions only indicate the possibility of frequency error, and not definitively so, is due to the case of a frequency-locked but jittery clock. Jitter can cause the filtered  $S_n$  to drift between zones, at times even causing 3-2-1 or 1-2-3 transitions. These transitions could mistakenly indicate frequency error. As described later, the proposed scheme filters these raw indicators to ensure the FD distinguishes frequency error from such jitter events.

### B. CDR Architecture

The proposed CDR architecture to achieve the above frequency acquisition scheme is shown in Fig. 8. The baud-rate architecture from [13] is extended to include: 1) a zero-level front-end comparator that feeds samples into the pattern filter; 2) the proposed adjustable baud-rate PD; and 3) the proposed FD. The FD consists of two components: the frequency detection logic to determine the direction of frequency error and the frequency correction logic to determine the change in the adjustable baud-rate PD to correct for this error.

Fig. 9 shows the state diagrams used to implement the transition detection. Two separate state machines, one for detecting slow clocks (Fig. 9(a)) and another for detecting fast clocks (Fig. 9(b)) monitor the valid  $S_n$  samples for changes in voltage zones. If the desired zone transitions are observed the state machine logic asserts its output signal. The raw outputs of these state machines are then used as slow and fast “triggers” in the frequency detection logic, as shown in Fig. 10. While the triggers are sent to the downstream frequency correction logic, the triggers are also accumulated by digital integrators. These accumulators are reset periodically based on a programmable period. The accumulated values, the slow and fast counts, are sent to the downstream correction logic as well for use by a frequency lock detector. This reset interval defines the window after which the lock detector determines if significant frequency error exists.

The frequency correction logic is shown in Fig. 11. The lock detector, within one observation window, observes the total and differential activities in the accumulated slow and fast counts through the  $CNT_{SUM}$  and  $CNT_{DIFF}$  signals, respectively. If either of these signals exceeds its respective threshold ( $N_{SUM}$  and  $N_{DIFF}$ ) within the reset interval, then the lock detector de-asserts  $FD_{LOCK}$ . As discussed in Section V, the threshold value  $N_{DIFF}$  plays a crucial role in the measured FD capture range.

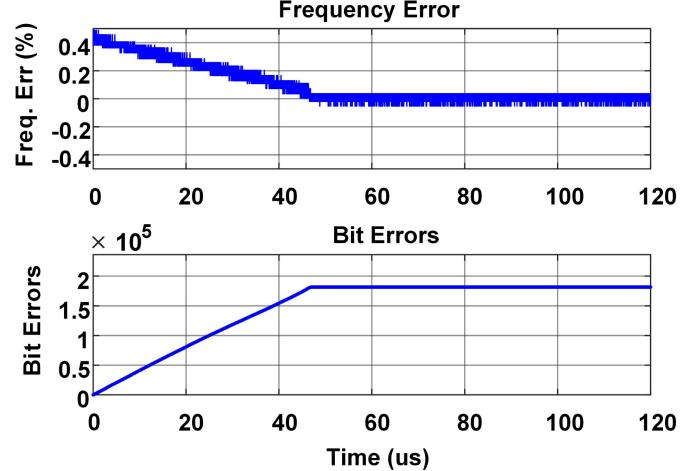


Fig. 13. System-level simulations of frequency error and bit error rate versus time with initial 0.5% frequency error.

If the lock detector determines frequency lock has not been achieved, the PD may require adjustment to correct for potential frequency error. A saturating integrator accumulates the difference between the raw state machine slow/fast trigger signals, and continues to accumulate for as long as FD lock is not achieved. As the signals are integrated, they indicate if the frequency error is too slow or too fast. If this integrated value exceeds its own threshold values ( $\pm FD_{TH}$ ), then the PD is adjusted to correct for the frequency error. If there is insufficient activity to exceed the thresholds, the PD remains in its normal operating mode. Crucially, when the lock detector determines frequency lock is achieved, the PD returns to its normal operating mode. The CDR then resumes normal phase correction on the frequency-locked clock.

Fig. 12 shows the conceptual functional behavior of the proposed frequency acquisition scheme. In the presence of initial frequency error, the accumulated signal from the frequency detection logic exceeds the detection threshold  $N_{DIFF}$  within one reset interval, and thus, the lock detector maintains a de-asserted  $FD_{LOCK}$ . The frequency correction logic then considers the integrated value  $FD_{INT}$  to determine which direction the PD should be adjusted. As  $FD_{INT}$  crosses its upper threshold, this indicates that the recovered clock is too slow, and hence the PD output characteristic is adjusted to correct for this slow clock over time. The PD continues to correct the frequency error until the lock detector asserts  $FD_{LOCK}$ , at which point the PD returns to normal operation.

The decreasing  $CNT_{DIFF}$  illustrates a case where the initial frequency error is relatively small. In such a case, as the adjusted PD corrects the frequency error, the peak  $CNT_{DIFF}$  at the end of each reset interval decreases: fewer 1-2-3 or 3-2-1 zone transitions are observed as the small frequency error is corrected. However, for larger frequency errors, the frequency offset occasionally causes the filtered sample  $S_n$  to skip zones. As the frequency error increases, this occurs more frequently. This results in a decreased number of transitions observed as frequency error becomes very large. This dependence on the frequency error limits the capture range of the FD, as will be discussed in Section V. Thus, for an initial large

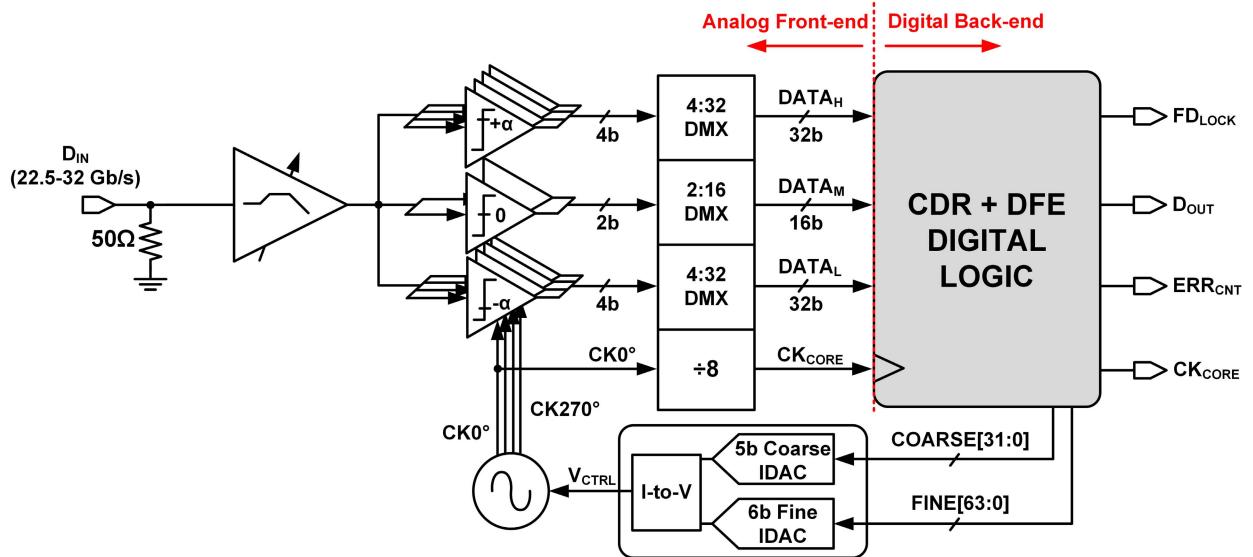


Fig. 14. Quarter-rate circuit implementation of the proposed receiver.

frequency error, the peak  $CNT_{DIFF}$  increases at first before decreasing as the frequency error is corrected.

Fig. 13 shows the simulated transient signals of the frequency error and bit error rate during CDR operation. The system-level simulation transmitted 28-Gb/s PRBS-31 data across a model of a Tyco 5" backplane channel with a 12-dB loss at 14 GHz and a model representing an *RC*-extracted CTLE equalizing the channel up to 7 GHz (sufficient for a one-tap DFE). The frequency error was initialized to  $+0.5\%$ . The frequency error was corrected by the CDR in  $50\ \mu s$ , after which point the FD achieved lock and bit errors were stabilized.

#### IV. IMPLEMENTATION

The full circuit implementation of the proposed receiver is shown in Fig. 14. The receiver was implemented as a quarter-rate architecture, where the nominal DCO frequency operates at one-quarter that of the equivalent full-rate clock (i.e.,  $f_{DCO} = f_{CK}/4$ ). Since baud-rate sampling requires one sample per UI, four phases of the quarter-rate clock are needed (CK 0°, CK 90°, CK 180°, and CK 270°). These clocks feed into a bank of ten double-tail comparators [20]. After the comparators sample the post-CTLE waveform, the samples are demuxed and sent to the synthesized digital core logic that realizes the FD logic, the DFE logic, and the CDR logic. The loop filter from the digital CDR feeds into the DCO, closing the loop.

Of the ten front-end comparators, eight define the upper and lower comparator levels ( $\pm\alpha$ ), and are clocked at each phase of the quarter-rate clock. The remaining two comparators set the zero-level threshold. The number of zero-level comparators was observed to affect the capture range of the frequency detection scheme, as shown in the system-level simulation results in Fig. 15. This simulated open-loop FD response configures the CDR in open-loop, sets the DCO frequency constant, and observes the accumulated FD differential

activity ( $CNT_{DIFF}$ ) at the end of a reset interval (16 384 UI). This was compared against frequency error (%) defined by

$$f_{ERR}(\%) = \frac{f_{DATA} - f_{CK}}{f_{CK}} \times 100\% \quad (1)$$

where  $f_{DATA}$  is the data rate and  $f_{CK}$  is the equivalent full-rate clock frequency. The open-loop FD range depicted in Fig. 15 examines  $CNT_{DIFF}$  before the  $N_{DIFF}$  threshold is applied in the FD logic (Fig. 11). The open-loop range here is defined as the range for which the sign of  $CNT_{DIFF}$  correctly indicates the direction of frequency error; the threshold for comparison for this figure is implicitly zero. This is to observe the maximum possible range for which the FD is operational. A non-zero  $N_{DIFF}$  reduces this range. However, a non-zero  $N_{DIFF}$  is required to assert  $FD_{LOCK}$  correctly, as discussed in Section V.

Increasing to four zero-level comparators (for a total of 12 comparators) increased the open-loop FD range but required additional comparators; reducing to one zero-level comparator (total of 9 comparators) significantly reduced the FD range. Two zero-level comparators (total of 10 comparators) was, therefore, chosen as a reasonable design tradeoff. It should be noted that the threshold  $N_{DIFF}$  is not necessarily the same for all three cases. As discussed in Section V,  $N_{DIFF}$  must be set to ensure correct  $FD_{LOCK}$  assertion. The 12 comparators' FD output is higher than the 10 comparators' output at the PD capture range boundary, as shown in Fig. 15(b), and therefore requires a higher  $N_{DIFF}$  value. Because of this, the use of 12 comparators offers negligible advantage in terms of capture range.

The CTLE consists of a source-degenerated stage, with a digitally programmable source-degeneration resistor, followed by a buffer stage. The CTLE response is sufficient to equalize the channel response up to  $f_{DATA}/4$ , as shown in Fig. 16, which is sufficient for the one-tap DFE. The DCO was implemented as a four-stage CML ring VCO controlled by coarse and fine current DACs, as shown in Fig. 17. The

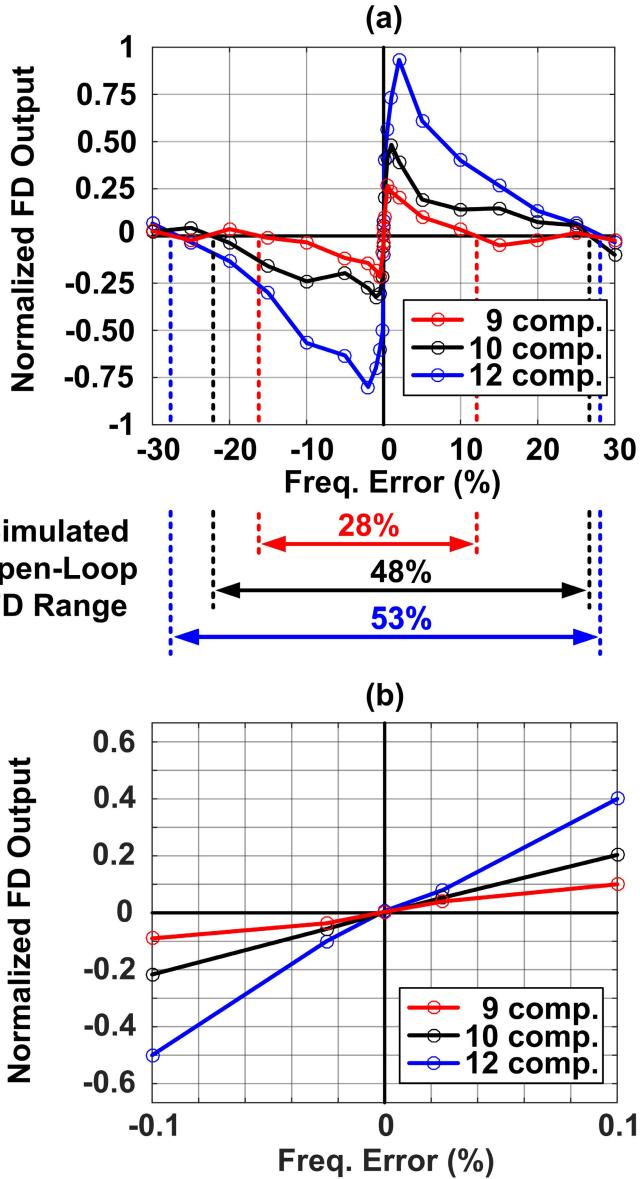


Fig. 15. System-level simulation comparing the open-loop FD response (PRBS-31) using nine (one zero-level comparator), ten (two zero-level comparators), and twelve (four zero-level comparators) total comparators; (b) Zoomed to  $\pm 0.1\%$   $f_{err}$  (PD capture range).

fine DAC was designed to span approximately two coarse codes. While both the coarse and fine codes change during acquisition, only the fine code needs to change to track during phase lock by nature of the rollover logic in the digital loop filter, as shown in Fig. 18. As changes to only the fine code and not the coarse code are required to maintain phase lock, the exact relationship between the fine DAC and coarse DAC is not critical; the fine DAC approximately spanning two coarse codes is sufficient for acquisition purposes.

## V. MEASUREMENT RESULTS

The proposed receiver was fabricated in TSMC's 28-nm HPM CMOS process. The die micrograph, along with the chip area of each block, is shown in Fig. 19.

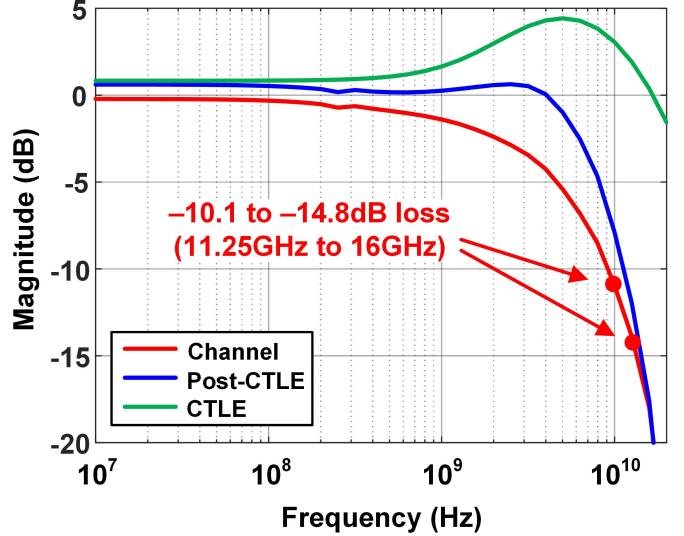


Fig. 16. Post-layout AC simulation of CTLE and post-CTLE response with measured total channel loss (Tyco 5" backplane and 6' SMA cables).

The total power consumption of the receiver operating at 32 Gb/s is 102.04 mW, excluding power associated with the I/O. The comparators, the associated high-speed clock tree, and the demux together consume 39.06 mW. The DCO, consisting of the VCO and its associated DAC, consumes 14.47 mW, while the CTLE consumes 14.43 mW. The synthesized digital core consumes 34.08 mW. The overall energy efficiency of the receiver at 32 Gb/s is 3.19 pJ/b. At the lower operating data rate, 22.5 Gb/s, the receiver consumes 65.2 mW, achieving 2.90-pJ/b efficiency.

The measurement setup is shown in Fig. 20. A transmitter clock source (Agilent N4960A) drives the transmit data pattern generator (Agilent N4951B). The pattern generator is limited to a maximum data rate of 32 Gb/s. The differential data signal is transmitted across a Tyco 5" backplane channel as well as 6' of SMA cables. The total measured channel frequency characteristic is shown in Fig. 16. The channel output is then wafer-probed onto the high-speed data input pads on the design under test. The high-speed differential recovered clock output is wafer-probed from the chip for analysis by a spectrum analyzer (Agilent N9010A). The remaining low-speed I/O are wire-bonded to a quad-flat no-leads package and soldered to a custom printed circuit board. The low-speed digital divided clock and two digital outputs are routed to a real-time oscilloscope (Agilent DSA-X 91604A). The digital outputs are programmed to display desired internal signals. A microcontroller (Arduino Uno) programs the receiver.

Fig. 21 shows the measured DCO response. For this measurement, the CDR is configured in open loop, and the digital fine and coarse codes are swept. The high-speed clock frequency is measured using the spectrum analyzer. The DCO output frequency ranges from 5.5 to 9 GHz. The measurement results indicate that the upper half of the DCO range for a given coarse code overlaps with lower half of the consecutive coarse code, verifying the fine DAC spans two coarse codes. For figure clarity, only every fourth coarse code is shown.

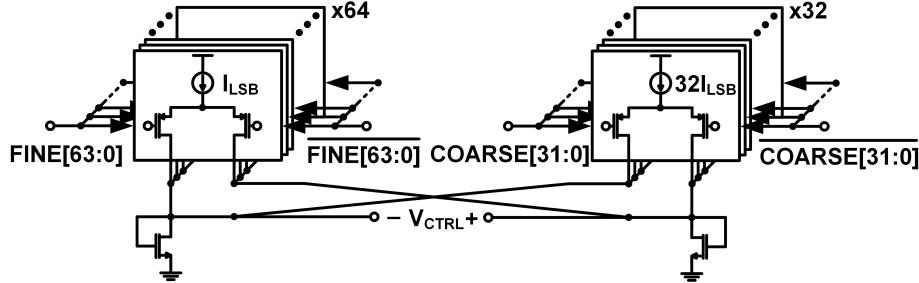


Fig. 17. Coarse and fine DACs for the DCO.

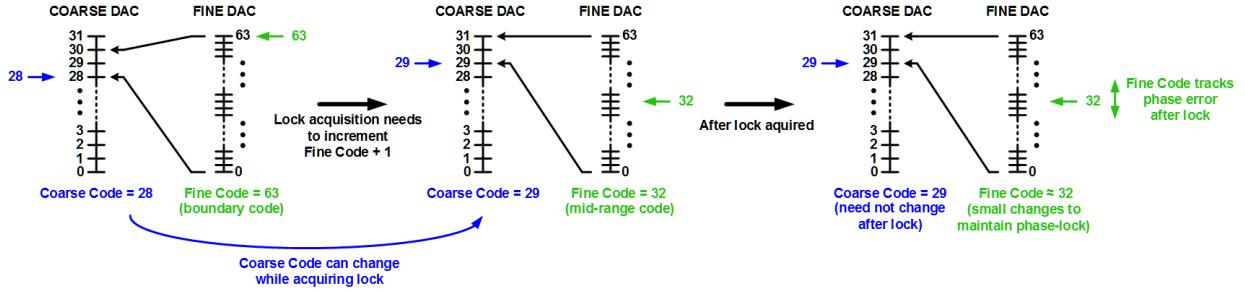


Fig. 18. Rollover logic for coarse and fine DAC codes.

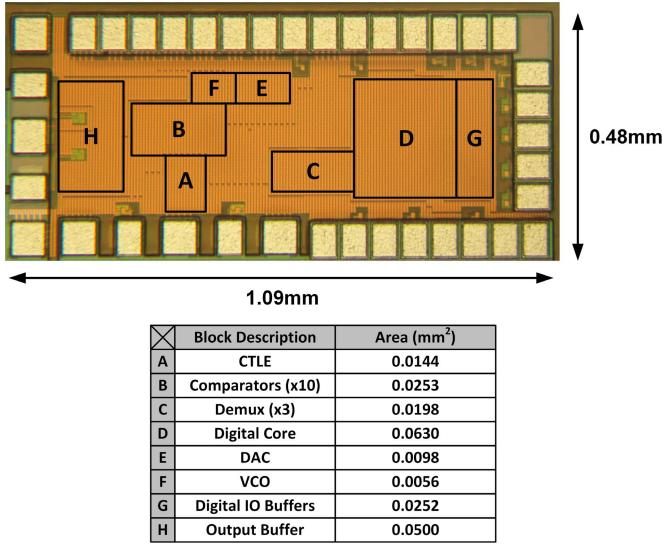


Fig. 19. Die micrograph.

Fig. 22 shows the measured and simulated open-loop FD response for PRBS-31 data input as a function of frequency error defined by (1). This experiment is conducted by configuring the CDR and the frequency acquisition in open loop, setting the DCO frequency constant. The DCO frequency is verified using the spectrum analyzer to ensure  $f_{DCO} = f_{CK}/4$  is as close to 7 GHz as possible. The data rate is swept relative to this frequency, and the accumulated FD differential activity ( $CNT_{DIFF}$ ) is observed at the end of a reset interval (16 384 UI). In this way, the FD response is measured up to  $f_{DATA} = 32$  Gb/s, the pattern generator's maximum data rate. This corresponds to  $f_{ERR} \leq 14.3\%$ , as shown in

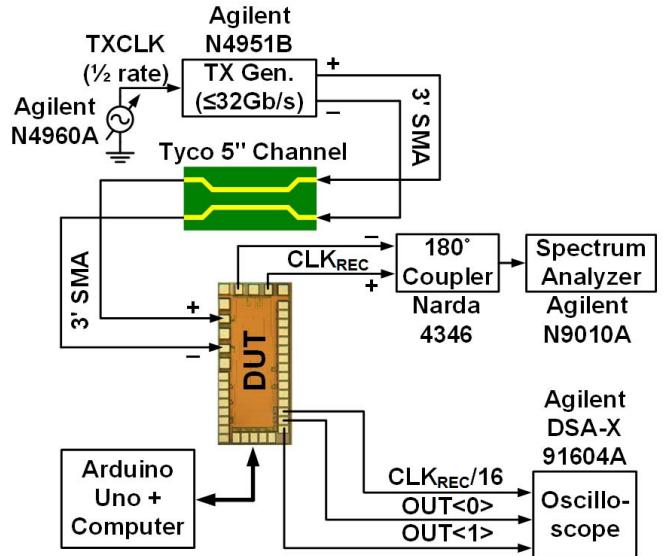


Fig. 20. Measurement setup.

interval A of Fig. 22. To measure beyond this (interval B of Fig. 22), the data rate is set to the maximum 32 Gb/s and  $f_{CK}$  is reduced. The FD output is normalized to its maximum observed value. The threshold values ( $\pm N_{DIFF} = \pm 32$  or  $\pm 0.2$  normalized) were set to ensure that  $FD_{LOCK}$  is asserted when the frequency error is nearly zero and within the PD capture range (approximately ±1000 parts per million (ppm)). This threshold, however, limits the FD open-loop capture range at large frequency errors. As frequency error increases in magnitude, filtered voltage zones can be skipped, and thus the FD observes the desired zone transitions 1-2-3 or 3-2-1 less frequently. For large frequency errors, these transitions are

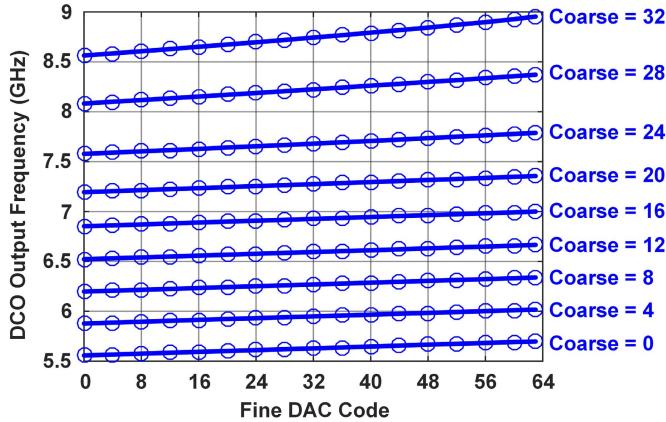


Fig. 21. Measured DCO response. For clarity, measurements from every fourth coarse code are displayed.

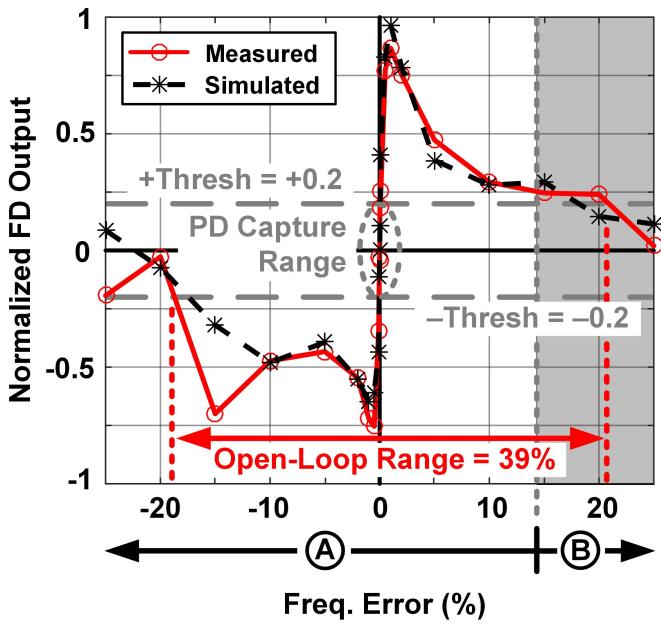


Fig. 22. Normalized measured versus simulated open-loop FD response for PRBS-31 data input. See text for measurement techniques in intervals A and B.

so infrequent that they fall below the threshold values; the FD is unable to confirm the presence of frequency error. The FD open-loop range is then defined as the range over which the FD output exceeds the threshold values, and places an upper bound on the expected closed-loop CDR capture range. Note that the threshold value  $N_{\text{SUM}}$  is measured not to have an observable effect, and thus, the threshold level is defined solely by  $N_{\text{DIFF}}$ . The measured open-loop FD range is 39%, and closely matches simulations.

Fig. 23 shows the measured capture range of the CDR operating in closed-loop. This capture range is measured by initializing the DCO to 7 GHz ( $f_{\text{CK}} = 28$  GHz) and measuring the widest range of data rates for which the CDR achieved lock. The capture range measurement was limited by the DCO lower frequency limit and the transmitter equipment upper frequency limit. When no TX jitter is applied, the CDR capture range is 34% with the FD enabled. Compared with

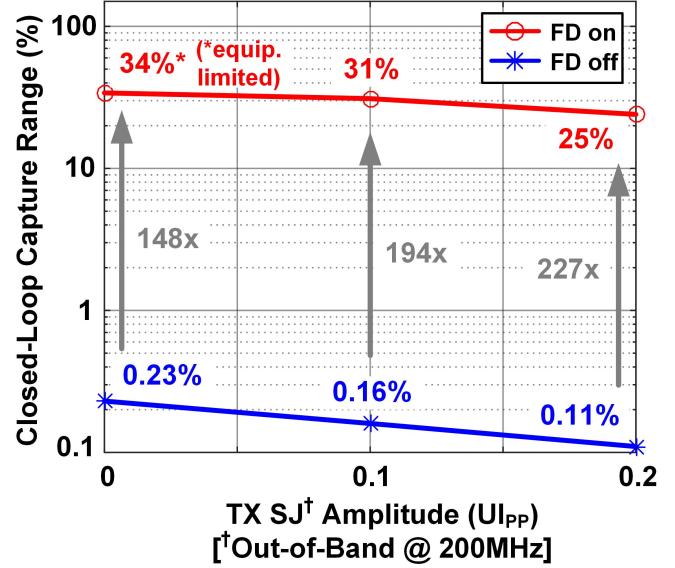


Fig. 23. Measured CDR capture range versus TX jitter (FD on versus FD off, PRBS-31).

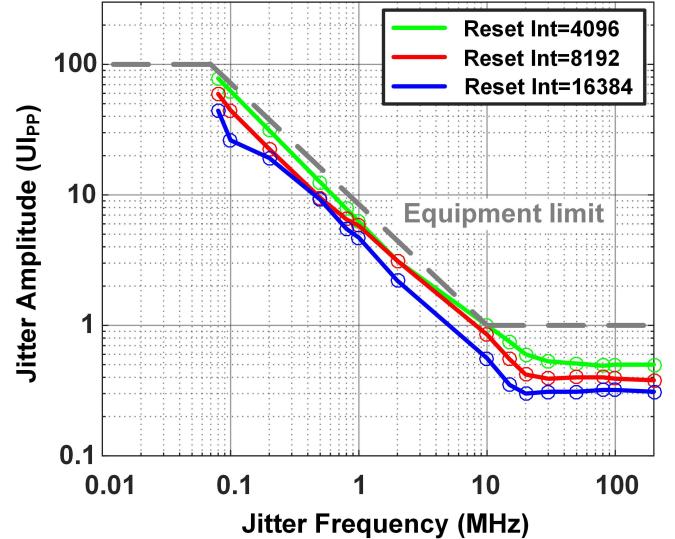


Fig. 24. Measured tolerance of  $FD_{\text{LOCK}}$  to TX sinusoidal jitter (28 Gb/s, PRBS-31, jitter applied after frequency acquisition).

the measured capture range of 2300 ppm (0.23%) with the FD disabled, the proposed frequency acquisition scheme results in a 148× increase in capture range. The capture range is also measured with 0.1 and 0.2 UI<sub>PP</sub> sinusoidal jitter (SJ) applied at 200 MHz. The capture range with the FD decreases to 25% as TX jitter is applied. Compared with no FD, however, the improvement is 227×.

Jitter impacts not only the frequency capture range but also lock detection. The measured ability of the frequency lock detector to tolerate TX SJ is shown in Fig. 24. Once the CDR achieves lock for 28-Gb/s PRBS-31 input data, TX jitter is applied until  $FD_{\text{LOCK}}$  is no longer asserted, measured to an error rate of  $10^{-12}$ . As shown, the key parameter affecting FD lock tolerance to jitter is the FD reset interval. As the reset interval increases and the FD observes longer UI sequences, jitter events that appear as 1-2-3 or 3-2-1 zone transitions

TABLE I  
COMPARISON WITH PREVIOUS WORKS

	JSSC 2013 [19]	ISSCC 2014 [15]	ISSCC 2014 [18]	JSSC 2015 [16]	This work
Technology	65nm CMOS	65nm CMOS	0.18μm BiCMOS	65nm CMOS	<b>28nm CMOS</b>
Supply Voltage	1.0	1.2/1.0	1.8	N/A	<b>0.9</b>
Baud-rate?	No	No	No	No	<b>Yes</b>
FD Type	DQFD	BBPD	Linear PD	Phase Selection	<b>Voltage-based</b>
Data rate (Gb/s)	8.5-11.5	4-10.5	8.2-10.3	8.5-12.1	<b>22.5-32*</b>
Capture Range	30%	65%	21%	36%	<b>34%</b>
Lock Time	< 400μs	<600μs	Not reported	Not reported	<b>&lt; 10.1ms</b>
Channel Loss (dB)	16.2	None reported	None reported	7.0	<b>14.8</b>
Equalization	CTLE + LA	None	None	None	<b>CTLE + 1-tap DFE</b>
Total Power (mW)	60** @ 11.5Gb/s	22.5 @ 10Gb/s	174 @ 10.3Gb/s	43.0 @ 12.1Gb/s	<b>102.0 @ 32Gb/s</b>
FoM (pJ/b)	5.22	2.25	16.89	3.55	<b>3.19</b>

\*Equipment limit (maximum data rate = 32Gb/s)

\*\*CDR power only (excludes CTLE and LA power)

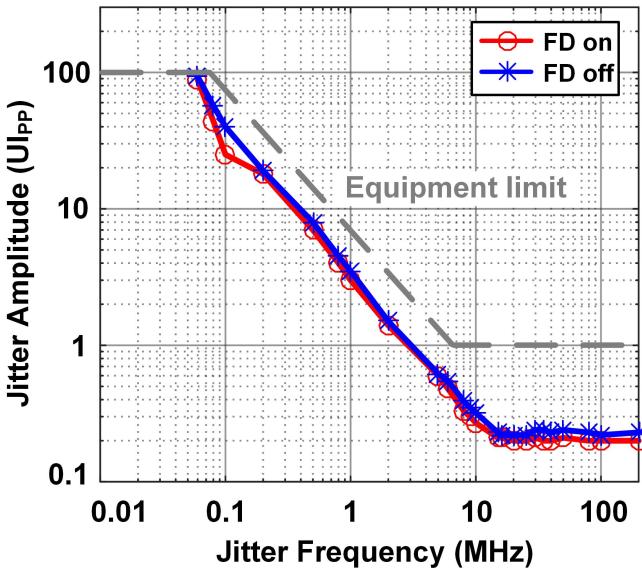


Fig. 25. Measured CDR jitter tolerance (28 Gb/s, PRBS-31, jitter applied after frequency acquisition).

accumulate and erroneously signal to the FD of possible frequency error. As the TX jitter increases, these events occur more frequently. If these events exceed the sum ( $N_{\text{SUM}}$ ) and differential ( $N_{\text{DIFF}}$ ) thresholds during a reset interval,  $\text{FD}_{\text{LOCK}}$  is de-asserted and erroneously activates the FD. To ensure the FD does not affect CDR jitter performance, the reset interval of 16 384 UI is chosen, such that the FD lock detector's ability to tolerate jitter exceeds the measured CDR jitter tolerance, as shown in Fig. 25.

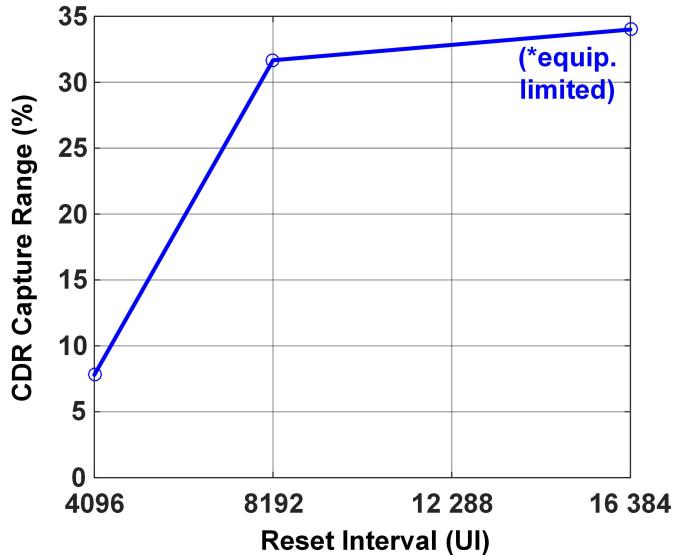


Fig. 26. Measured CDR capture range versus reset interval (equipment limit due to maximum 32-Gb/s data rate of pattern generator).

The reset interval also impacts capture range. For fixed  $N_{\text{SUM}}$  and  $N_{\text{DIFF}}$  threshold values, the measured capture range as a function of the reset interval is shown in Fig. 26. Compared with Fig. 24 previously discussed, decreasing the reset interval increases the jitter performance but drastically reduces the capture range. The FD is designed to maximize capture range while ensuring the FD's tolerance to jitter exceeds that of the CDR. Therefore, a reset interval

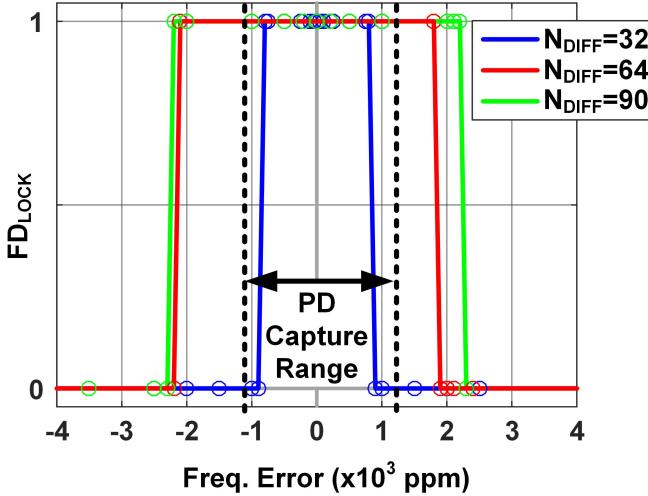


Fig. 27. Measured FD<sub>LOCK</sub> versus frequency error (open loop,  $N_{\text{SUM}} = 80$ ).

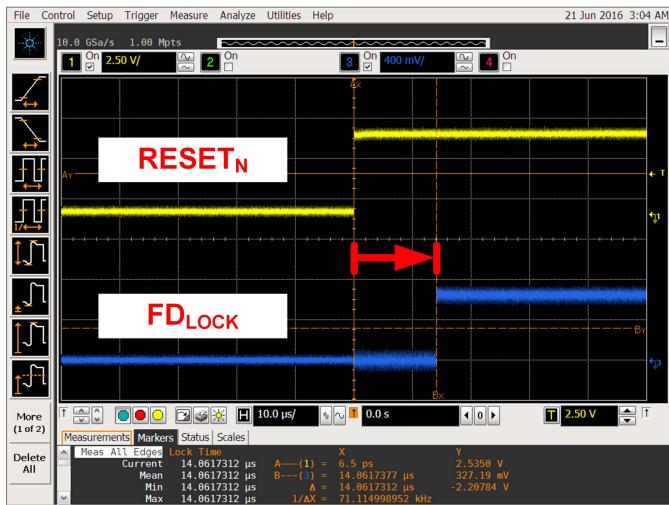


Fig. 28. Example measurement of CDR lock time.

of 16 384 UI is chosen to achieve the maximum measured capture range.

The interaction between the frequency correction and the PD is summarized in Fig. 27. The lock detector differential threshold  $N_{\text{DIFF}}$  was observed to affect the frequency error range in which the lock detector asserted FD<sub>LOCK</sub>. The threshold of  $N_{\text{DIFF}} = 32$  ensured frequency lock was asserted within the measured capture range of the PD. As the FD approaches frequency lock, an analytical relationship between  $N_{\text{SUM}}$  and  $N_{\text{DIFF}}$  and the frequency error can be found with respect to ISI. Based on the reset interval and the average slope in the data waveform due to ISI, minimum  $N_{\text{DIFF}}$  and  $N_{\text{SUM}}$  values are required to target a desired residual frequency error. This can be derived as

$$|f_{\text{ERR,LOCK}}| < \frac{\min(N_{\text{SUM}}, N_{\text{DIFF}})}{N_{\text{RST}}} \frac{2\alpha}{2 - \alpha} \quad (2)$$

where  $N_{\text{RST}}$  is the reset interval length and  $|f_{\text{ERR,LOCK}}|$  is the normalized frequency error when FD<sub>LOCK</sub> is asserted. This closely matches the measured lock ranges in Fig. 27

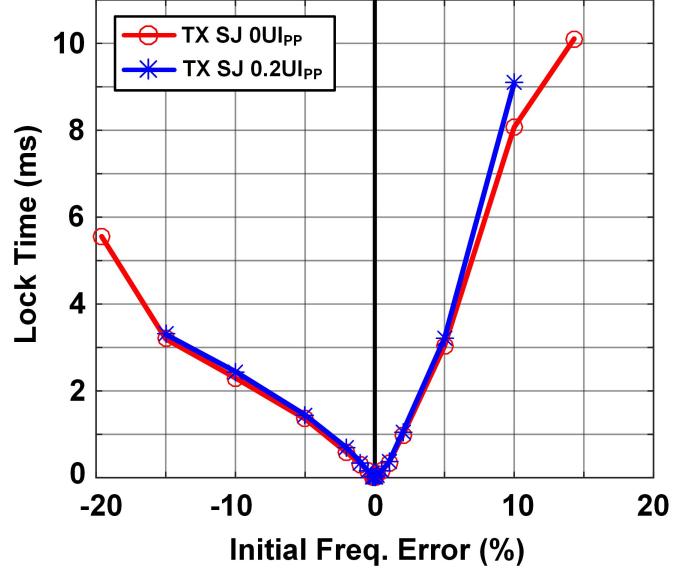


Fig. 29. Measured CDR lock time versus initial frequency error.

at 28 Gb/s ( $\alpha = 0.39$ ). It should be noted that the “011” and “100” data transition densities have no effect in (2);  $N_{\text{DIFF}}$  and  $N_{\text{SUM}}$  are effectively normalized against transition density.

Finally, the lock time of the CDR with the proposed FD is presented. The lock time measurement is conducted using the real-time oscilloscope, observing the CDR reset signal and the FD<sub>LOCK</sub> signal on the configurable digital outputs, as shown in Fig. 28. The time between the de-assertion of the reset signal and the assertion of FD<sub>LOCK</sub>, i.e., the lock time, is plotted in Fig. 29 in the cases of no TX SJ jitter and 0.2 UI<sub>PP</sub> TX SJ jitter at 200 MHz. A maximum lock time of 10.1 ms is measured.

Table I compares this paper to previous works in referenceless CDRs based on digital quadri-correlators [19], bang-bang and linear PDs [15], [18], and phase selection [16]. This paper is the first referenceless CDR to operate at baud rate. It achieves the highest reported data rate with the widest absolute capture range of 9.5 Gb/s. It is the first to be designed with channel loss in mind, achieving operation from 10.1 down to 14.8-dB loss at Nyquist ( $\alpha$  ranging from 0.26 to 0.51) with the help of a CTLE and a one-tap DFE.

## VI. CONCLUSION

This paper presents the first reported baud-rate referenceless CDR. In conjunction with a CTLE and a one-tap DFE, it receives 22.5–32 Gb/s PRBS-31 data transmitted across a Tyco 5" channel with Nyquist loss ranging from 10.1 dB to 14.8 dB. Unlike existing referenceless CDRs, the proposed frequency acquisition scheme does not require an additional frequency-locked loop in the CDR, is able to tolerate data waveforms with significant ISI by design, and conducts frequency correction by adjusting the digital PD characteristics, such that the average PD output corrects the frequency error. The receiver, with CTLE and DFE, achieved energy consumption below 3.2 pJ/b.

## REFERENCES

- [1] J. D. H. Alexander, "Clock recovery from random binary data," *Electron. Lett.*, vol. 11, no. 22, pp. 541–542, Oct. 1975.
- [2] J. Savoj and B. Razavi, "A 10-Gb/s CMOS clock and data recovery circuit with a half-rate binary phase/frequency detector," *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 13–21, Jan. 2003.
- [3] N. Nedovic *et al.*, "A 40–44 Gb/s 3× oversampling CMOS CDR/1:16 DEMUX," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2726–2735, Dec. 2007.
- [4] S. Shekhar, R. Inti, J. Jaussi, T.-C. Hsueh, and B. Casper, "A 1.2–5 Gb/s 1.4–2 pJ/b serial link in 22 nm CMOS with a direct data-sequencing blind oversampling CDR," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, Kyoto, Japan, Jun. 2015, pp. C350–C351.
- [5] M. Brownlee, P. K. Hanumolu, and U.-K. Moon, "A 3.2 Gb/s oversampling CDR with improved jitter tolerance," in *Proc. IEEE Custom Integr. Circuits Conf.*, San Jose, CA, USA, Sep. 2007, pp. 353–356.
- [6] J.-M. Lin, C.-Y. Yang, and H.-M. Wu, "A 2.5-Gb/s DLL-based burst-mode clock and data recovery circuit with 4× oversampling," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 23, no. 4, pp. 791–795, Apr. 2015.
- [7] K. Mueller and M. Müller, "Timing recovery in digital synchronous data receivers," *IEEE Trans. Commun.*, vol. COM-24, no. 5, pp. 516–531, May 1976.
- [8] F. Spagna *et al.*, "A 78 mW 11.8 Gb/s serial link transceiver with adaptive RX equalization and baud-rate CDR in 32 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2010, pp. 366–367.
- [9] A. K. Joy *et al.*, "Analog-DFE-based 16 Gb/s SerDes in 40 nm CMOS that operates across 34 dB loss channels at Nyquist with a baud rate CDR and 1.2 V<sub>pp</sub> voltage-mode driver," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2011, pp. 350–351.
- [10] C. Ting, J. Liang, A. Sheikholeslami, M. Kibune, and H. Tamura, "A blind baud-rate ADC-based CDR," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2013, pp. 122–123.
- [11] P. A. Francese *et al.*, "A 16 Gb/s 3.7 mW/Gb/s 8-tap DFE receiver and baud-rate CDR with 31 kppm tracking bandwidth," *IEEE J. Solid-State Circuits*, vol. 49, no. 11, pp. 2490–2502, Nov. 2014.
- [12] R. Dokania *et al.*, "A 5.9 pJ/b 10 Gb/s serial link with unequalized MM-CDR in 14 nm tri-gate CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2015, pp. 184–185.
- [13] T. Shibasaki *et al.*, "A 56 Gb/s NRZ-electrical 247 mW/lane serial-link transceiver in 28 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2016, pp. 64–65.
- [14] M. S. Jalali, R. Shivnaraine, A. Sheikholeslami, M. Kibune, and H. Tamura, "An 8 mW frequency detector for 10 Gb/s half-rate CDR using clock phase selection," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, San Jose, CA, USA, Sep. 2013, pp. 1–4.
- [15] G. Shu, W.-S. Choi, S. Saxena, T. Anand, A. Elshazly, and P. K. Hanumolu, "A 4-to-10.5 Gb/s 2.2 mW/Gb/s continuous-rate digital CDR with automatic frequency acquisition in 65 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2014, pp. 150–151.
- [16] M. S. Jalali, A. Sheikholeslami, M. Kibune, and H. Tamura, "A reference-less single-loop half-rate binary CDR," *IEEE J. Solid-State Circuits*, vol. 50, no. 9, pp. 2037–2047, Sep. 2015.
- [17] W. Rahman *et al.*, "A 22.5-to-32 Gb/s 3.2 pJ/b referenceless baud-rate digital CDR with DFE and CTLE in 28 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2017, pp. 120–121.
- [18] S. Huang, J. Cao, and M. Green, "An 8.2-to-10.3 Gb/s full-rate linear reference-less CDR without frequency detector in 0.18 μm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2014, pp. 152–153.
- [19] N. Kocaman *et al.*, "An 8.5–11.5-Gbps SONET transceiver with referenceless frequency acquisition," *IEEE J. Solid-State Circuits*, vol. 48, no. 8, pp. 1875–1884, Aug. 2013.
- [20] D. Schinkel, E. Mensink, E. Klumperink, E. van Tuijl, and B. Nauta, "A double-tail latch-type voltage sense amplifier with 18 ps setup+hold time," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2007, pp. 314–315.



**Wahid Rahman** was born in Muroran, Japan, and grew up in Ottawa, Canada. He received the B.A.Sc. degree (Hons.) in engineering science and the M.A.Sc. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2014 and 2017, respectively.

He was an Undergraduate Design Intern with Altera, Toronto, ON, Canada, from 2012 to 2013, where he was involved in modeling 20-nm PLLs, and a Graduate Research Intern with Fujitsu Laboratories Ltd., Kawasaki, Japan. Later in 2017, he joined AlphaWave IP, Toronto, ON, Canada, as a Senior Engineer working on research and development of high-speed serial interfaces.

Mr. Rahman received the Governor General's Bronze Academic Medal, the Ontario Graduate Scholarship, the NSERC Canada Graduate Scholarship, and the 2017 Analog Devices Outstanding Student Designer Award.



**Danny Yoo** received the B.A.Sc. degree in electrical engineering from the University of Toronto, Toronto, ON, Canada, in 2015, where he is currently pursuing the M.A.Sc. degree in electrical engineering, with a focus on clock and data recovery for high-speed wireline communication.

From 2013 to 2014, he was an Analog Layout Designer with Intel Corporation, Toronto, ON, Canada, where he was involved in high-speed SerDes IP.



**Joshua Liang** received the B.A.Sc. degree in engineering science and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of Toronto, Toronto, ON, Canada, in 2007, 2009, and 2017, respectively.

From 2009 to 2011, he was an Analog Designer with Microsemi, Ottawa, ON, Canada, where he was involved in circuits for low-jitter clock synthesis. Since 2012, he has been involved in research on wireline transceivers. In 2017, he joined Huawei, Toronto, ON, Canada, where he is involved in the research and development of high-speed interfaces.

Dr. Liang was a recipient of the Analog Devices' Outstanding Student Designer Award in 2016.



**Ali Sheikholeslami** (S'98–M'99–SM'02) received the B.Sc. degree from Shiraz University, Shiraz, Iran, in 1990, and the M.A.Sc. and Ph.D. degrees from the University of Toronto, Toronto, ON, Canada, in 1994 and 1999, respectively, all in electrical engineering.

In 1999, he joined the Department of Electrical and Computer Engineering, University of Toronto, where he is currently a Professor. He was on research sabbatical with Fujitsu Laboratories Ltd., Kawasaki, Japan, from 2005 to 2006 and Analog Devices, Toronto, from 2012 to 2013. He has co-authored over 50 journal and conference papers and eight patents. His current research interests include analog and digital integrated circuits, high-speed signaling, and VLSI memory design.

Dr. Sheikholeslami received numerous teaching awards, including the 2005–2006 Early Career Teaching Award and the 2010 Faculty Teaching Award from the Faculty of Applied Science and Engineering, University of Toronto. He served on the Memory, Technology Directions, and Wireline Subcommittees of the IEEE International Solid-State Circuits Conference (ISSCC) from 2001 to 2004, 2002 to 2005, and 2007 to 2013, respectively. He is currently the Educational Events Chair of the ISSCC and an Associate Editor of the *Solid-State Circuits Magazine*. He was an Associate Editor of the *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-PART I (TCAS-I)* from 2010 to 2012, and the Program Chair of the 2004 IEEE International Symposium on Multiple-Valued Logic. Since 2016, he has been the Education Chair and the Distinguished Lecturer Program Chair of the Solid-State Circuits Society and an Elected Member of its Administration Committee. He is a registered Professional Engineer in Ontario, Canada.



**Hirotaka Tamura** (M'02–SM'10–F'13) received the B.S., M.S., and Ph.D. degrees in electronic engineering from Tokyo University, Tokyo, Japan, in 1977, 1979, and 1982, respectively.

He joined Fujitsu Laboratories Ltd., Kawasaki, Japan, in 1982. After being involved in the development of different exploratory devices, such as Josephson junction devices and high-temperature superconductor devices, he moved into the field of CMOS high-speed signaling in 1996 and involved in the development of a multi-channel high-speed I/O for server interconnects. Since then, he has been involved in the area of architecture- and transistor-level design for CMOS high-speed signaling circuits.



**Hisakatsu Yamaguchi** received the B.S. degree in electrical engineering from the Tokyo University of Science, Chiba, Japan, in 1994, and the M.S. degree in electrical engineering from The University of Tokyo, Tokyo, Japan, in 1996.

In 1996, he joined Fujitsu Laboratories Ltd., Kawasaki, Japan, where he was involved in research on DRAMs with high-speed I/Os and was responsible for developing MPEG4 Codec ICs. He is currently involved in developing high-speed I/Os for high-end servers and super-computers.

Mr. Yamaguchi served on the technical program committees of the IEEE International Solid-State Circuits Conference from 2012 to 2016.



**Takayuki Shibasaki** received the B.S., M.S., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 2003, 2005, and 2008, respectively.

He joined Fujitsu Laboratories Ltd., Kawasaki, Japan, in 2008, where he is currently involved in high-speed CMOS interface circuit design.