7.4 Gb/s 6.8 mW Source Synchronous Receiver in 65 nm CMOS

Masum Hossain, *Member, IEEE*, and Anthony Chan Carusone, *Senior Member, IEEE*

***Abstract*—A high-frequency jitter tolerant receiver in 65 nm CMOS is presented. Jitter tolerance is improved by tracking correlated jitter using a pulsed clock forwarded from the transmitter side. The clock receiver comprises two injection locked oscillators to frequency-multiply, deskew, and adjust jitter tracking bandwidth. Different data rates and latency mismatch between the clock and data paths are accommodated by a jitter tracking bandwidth that is controllable up to 300 MHz. Each receiver consumes 0.92 pJ/bit operating at 7.4 Gb/s and has a jitter tolerance of 1.5 UI at 200 MHz.**

***Index Terms*—Injection locking, jitter tracking, source synchronous.**

I. Introduction

PARALLEL interfaces are becoming increasingly important to meet the aggregate bandwidth required between microprocessors, memory, peripheral components, network hubs, storage devices, etc. A combination of technology scaling and architectural innovation has significantly improved the power efficiency of these links over the years [Fig. 1(a)]. This summary includes both source synchronous and asynchronous links. For multilink high aggregate bandwidth interface source synchronous links are useful where their power consumption can be amortized across multiple links in the system. For example, both QPI and HyperTransport include a dedicated link to carry a synchronous clock from the transmitter to receiver and shared by 5–20 data transceivers. At 10+ Gb/s, timing margin is scarce and power supply induced jitter is considered the main source of jitter. The spectrum of this jitter is strongly correlated to the power supply distribution impedance, which peaks at the resonant frequency of on-die decoupling capacitance and bond wire inductance. In most digital systems this resonant frequency varies from 50 MHz to 400 MHz. For example, the next generation Intel core (Nehalem) supply network resonates at 300 MHz [1]. A similar impedance profile has been reported in other digital systems [2]. Therefore, it is desired that high-speed I/O receivers have higher jitter tolerance from 50 MHz to 400 MHz. One possible solution is to achieve higher tolerance by increasing the jitter tracking bandwidth of the receiver. Unfortunately most state-of-the-art clock and data recovery (CDR) units have a tracking bandwidth less than 10 MHz. Increasing the CDR bandwidth beyond 100 MHz can significantly increase power consumption and stress stability requirements.

Instead, a source synchronous receiver can track jitter on a source-synchronous clock forwarded from the transmitter as shown in Fig. 1(b). In these links static phase offset is corrected by per pin phase compensation loops. Per pin deskewing is done at startup [3]; the optimum deskew setting is stored and the calibration circuitry turned off during normal operation. In some cases the deskew loop remains active during the normal operation, but the tracking bandwidth is only on the order of kHz [4]. Dynamic phase errors (jitter) are tracked using the forwarded clock whose jitter is correlated to that of the data because both share the same frequency synthesizer and transmit circuitry. Hence, jitter tolerance is improved by retiming the data with a clock that tracks correlated jitter on the forwarded clock [5]. In an ideal scenario, both clock and data should appear at the receiver with the same latency, making the link tolerable to any jitter that appears in common on both clock and data. In reality, the latency of the clock and data paths are not perfectly matched. Assuming a latency mismatch ΔT, the jitter transfer function (JTF) of the forwarded clock path can be written as $e^{-(\Delta T)s}H_\Phi(s)$ where H_Φ is the jitter transfer function of any clocking circuitry in the clock path. Using this JTF, the jitter tolerance of a forwarded clock receiver can be written as

$$J_{\mathrm{TOL}}(s) = \left| \frac{0.5}{1 - e^{-(\Delta T)s}H_\Phi(s)} \right|. \tag{1}$$

For example, if a second-order PLL is included in the clock path, $H_\Phi = H_{\mathrm{PLL}} = (2\zeta\omega_n s + \omega_n^2)/(s^2 + 2\zeta\omega_n s + \omega_n^2)$. Alternatively, if a DLL appears in the clock path, $H_\Phi = H_{\mathrm{DLL}} = (1 + se^{-\tau s}/\omega_P)/(1 + s/\omega_P)$ where $\omega_P = I_P/2\pi(sC_P)$ is the pole introduced by the loop filter. The JTF and jitter tolerance for these two cases are compared in Fig. 2 with 5 UI and 1 UI of latency mismatch between the clock and data paths. In most high speed applications, the PLL bandwidth is on the order of 10–30 MHz. As a result, the link's jitter tolerance is reduced to 0.5 UI(peak) around 25 MHz. Note that the latency mismatch between clock and data paths has little effect on jitter tolerance when a PLL is used since high-frequency jitter is always filtered. A DLL on the other hand provides an all-pass JTF with small peaking (less than 1 dB). Assuming $H_{\mathrm{DLL}} \approx 1$, jitter tolerance using a DLL will be less than 0.5 UI if the denominator is greater than 1:

$$|1 - e^{-(\Delta T)s}| > 1. \tag{2}$$

Manuscript received September 10, 2010; revised January 12, 2011; accepted March 09, 2011. Date of current version May 25, 2011. This paper was approved by Associate Editor Jafar Savoj. This work was supported by Intel Corporation and Broadcom. Fabrication services were provided by CMC.

The authors are with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, M5S 3G4 Canada (e-mail: masum@eecg.utoronto.ca; tony.chan.carusone@isl.utoronto.ca).

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.2011.2131730

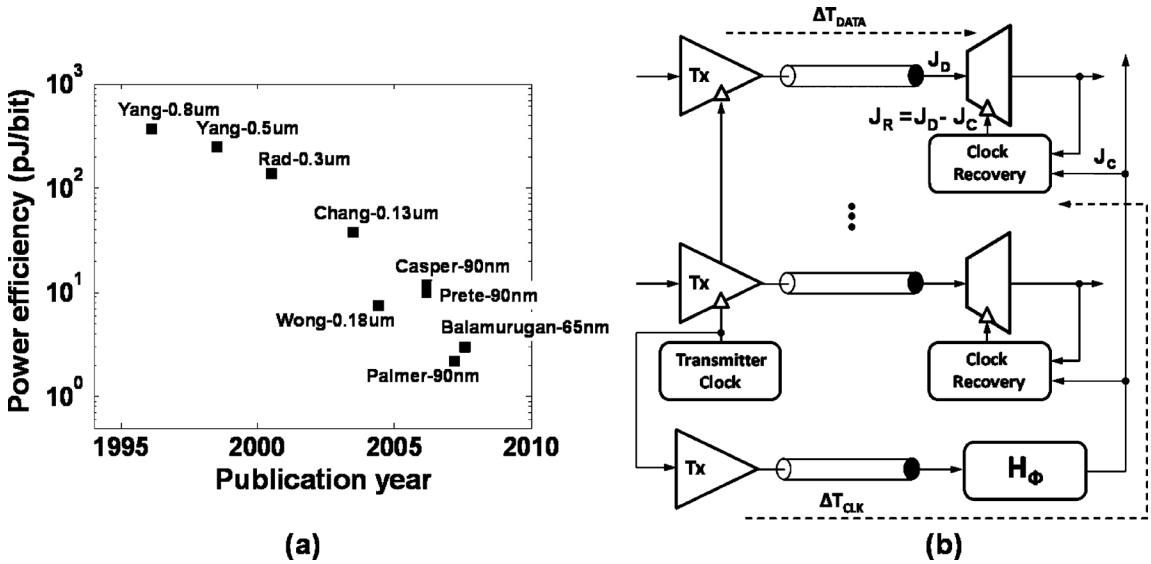


Fig. 1. (a) Low power transceiver power efficiency over the years. (b) Clock forwarded receiver architecture.

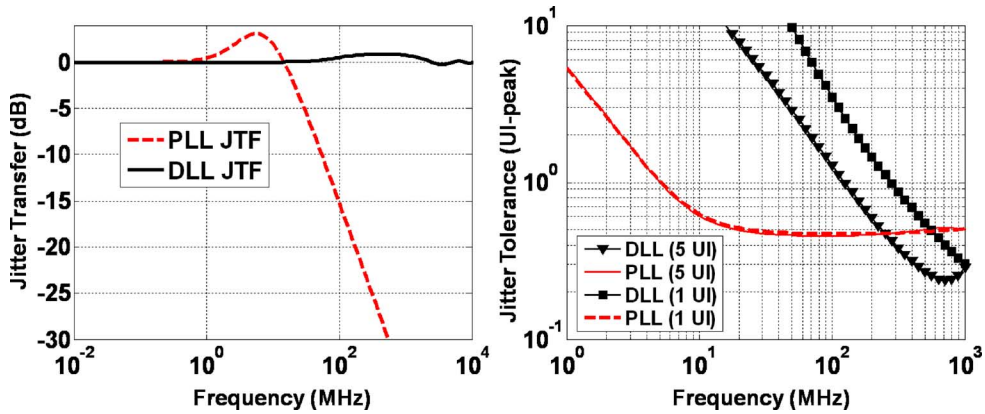


Fig. 2. Jitter transfer and jitter tolerance of clock forwarded receivers with 1 UI and 5 UI latency mismatch between clock and data. For the PLL, $\omega_n = 2\pi \times 7 \times 10^6$ rad/s, $\zeta = 1$, $f_{-3\mathrm{dB}} = 25$ MHz. For the DLL, $\omega_P = 2\pi \times 100 \times 10^6$ rad/s and $\tau = 250$ ps.

This condition can be further simplified to relate maximum allowable latency mismatch ΔT UI to jitter frequency f_{jitter} and bit rate, f_{bit}:

$$\Delta T < \frac{f_{\mathrm{bit}}}{8 f_{\mathrm{jitter}}}. \tag{3}$$

This condition simply states that for a given data rate DLL jitter tracking is effective as long as the latency mismatch between clock and data is less than $f_{\mathrm{bit}}/8f_{\mathrm{jitter}}$. At larger latency mismatches, correlated jitter tracking is detrimental since the phase shift between the jitter on the clock and data causes the correlated jitter to superimpose constructively. For example, using the above expression, for 100 MHz jitter, the maximum allowable latency mismatch is 12 UI. However, if the jitter frequency is 500 MHz, in that case the maximum allowable latency mismatch is only 2.5 UI. In summary, jitter tolerance using a DLL is strongly influenced by the latency mismatch and frequency of jitter [6], [7]. Low- and mid-frequency jitter which appears in-phase at the sampling clock is beneficial and should be tracked. But high-frequency jitter which appears out of phase due to latency degrades timing margin, and hence should be filtered.

II. Optimum Jitter Tracking

To improve jitter tolerance we replace the all-pass jitter transfer function with a first-order low-pass filter where the jitter tracking bandwidth (JTB) is varied. Consider a transmitted data signal is phase modulated with sinusoidal jitter $J_D = A\sin(\omega_j t)$ where A is the jitter amplitude and ω_j is the jitter frequency. The forwarded clock signal is also modulated with the same sinusoid jitter, but with added latency of M UI, $J_C = A\sin(\omega_j(t - MT))$. The jitter of the forwarded clock is shaped by a low-pass filter, where the low-pass jitter filter transfer function is $|H_\Phi(\omega_j)| = 1/\sqrt{1 + (\omega_j/\omega_P)^2}$. When normalized to the data signal jitter amplitude, the resulting effective jitter is

$$\frac{J_R}{J_D} = 1 - \frac{\sin(\omega_j(t - MT))}{\sin(\omega_j t)\sqrt{1 + \left(\frac{\omega_j}{\omega_P}\right)^2}}. \tag{4}$$

This theoretical expression of normalized jitter is plotted along with behavioral simulation results in Fig. 3(a) for a jitter frequency of $\omega_j = 2\pi \times 200$ MHz sinusoid. Similar results are obtained for 100 MHz and 300 MHz jitter in Fig. 3(b). For all

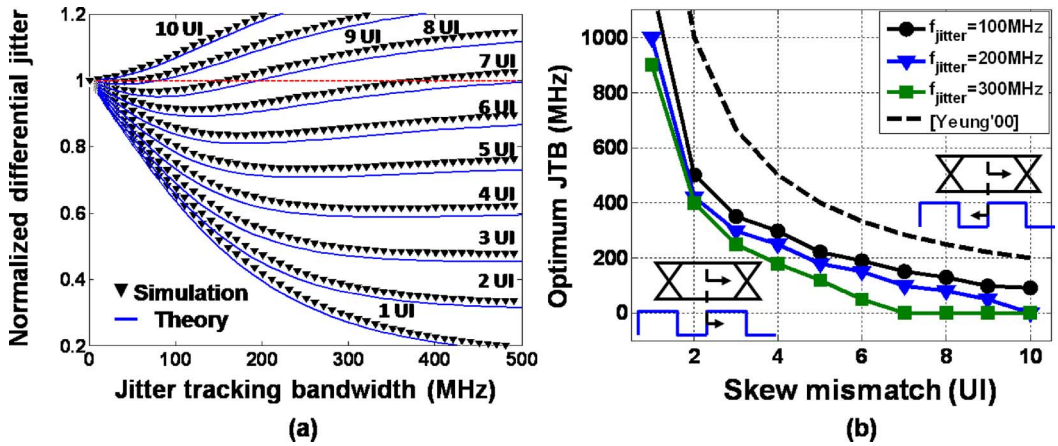


Fig. 3. (a) Normalized jitter amplitude as a function of jitter tracking bandwidth. (b) Optimum jitter tracking bandwidth as a function of the latency mismatch between the clock and data paths.

three jitter frequencies (100 MHz, 200 MHz, and 300 MHz) the optimum JTB decreases with increasing latency mismatch until eventually for 10 UI or higher latency mismatch there is no benefit of jitter tracking through the forwarded clock path. For less than 1 UI latency mismatch, best jitter tolerance is obtained with the highest possible jitter tracking bandwidth. However, it is very difficult to achieve this level of matching in practical clock forwarded systems. More realistic latency mismatch for a clock forwarded link varies from 2 to 6 UI for which the optimum jitter tracking bandwidth with respect to 100–300 MHz jitter varies from 400 MHz to 25 MHz.

Another important consideration is jitter amplification. In lossy, bandwidth limited channels forwarded clock jitter is amplified. To avoid jitter amplification a sub-rate forwarded clock is preferred. In that case, the forwarded clock must be frequency-multiplied and aligned with the data at each receiver. In summary, the clock path in a clock forwarded transceiver should provide flexible clock multiplication, a controlled phase shift, and a JTB adjustable over hundreds of MHz to accommodate different channel losses, supply resonance, bit rates, and path delay mismatches. In this work we propose dual phase filtering using injection locked oscillators that provides all of the above functionalities. First the fractional rate forwarded clock is multiplied up to a full rate clock which is distributed to all data lanes with a jitter tracking bandwidth adjustable from 25 MHz to 400 MHz. Each receiver accepts this differential distributed clock and generates any sampling phase between $0 \rightarrow 360$ while also providing another first-order jitter filter to further suppress uncorrelated high-frequency jitter.

III. Architecture Review

Source synchronous clock paths have been implemented with many different combinations of DLLs, PLLs and phase interpolator. A PLL can provide both clock multiplication and jitter filtering. The approach in Fig. 4(a) is to use cascaded PLLs: a shared low bandwidth PLL is used as a clock multiplying unit (CMU). This multiplying PLL (MPLL) generates a high-frequency clock from the forwarded sub-rate clock as shown in Fig. 4(a). Following that a local PLL (LPLL) is used in each lane to generates multiple clock phases for phase interpolation [8]. In existing MPLL implementations, the tracking bandwidth f_{p1} is much lower compared to LPLL bandwidth f_{p2}. A conventional PLL is a second-order system with a stabilizing zero. Its jitter tracking bandwidth is limited by stability requirements which must be ensured over process corners with temperature and supply variation. As a result, when used in a clock forwarded system, such a system can not obtain an overall jitter tracking bandwidth of hundreds of MHz and hence filters out useful correlated jitter [6].

To avoid filtering useful correlated jitter, most of the existing source synchronous links prefer DLLs over PLLs. For example, the QPI interface implemented in [1] uses a DLL to generate multiple clock phases. These clock phases are distributed to each receiver and then interpolated to generate required sampling phase between $0 \rightarrow 360$. Since at least four clock phases 0, 90, 180, and 270 are distributed in this approach, clock distribution network consumes significant power. Both the DLL and phase interpolator provide all-pass jitter transfer, so tracks both in-phase and out-of-phase jitter. Moreover, high-frequency jitter such as duty cycle distortion (DCD) is amplified. Therefore a DCD correction loop is often required in DLL based systems, as illustrated in Fig. 4(b) [6], [9]. Alternatively a phase filter can be implemented by "time averaging" after the DLL which further increases power consumption and complexity[1]. In addition the DLL does not provide flexible clock multiplication, so a full rate clock needs to be forwarded.

Injection locked oscillators (ILO) are a power- and area-efficient alternative to PLLs and DLLs. In [10], [11] an ILO performs both jitter filtering and clock deskew by introducing a frequency offset between the ILO's free-running frequency and the injected frequency. Note that no phase detector, charge pump or loop filter is required in this architecture. Therefore, excellent area and power efficiency is reported in [10], [11]. However, this simple architecture, illustrated in Fig. 4(c), has several limitations. First, the deskew range is limited and within the deskew range jitter tracking bandwidth varies significantly as a function of phase deskew setting. Theoretical limit of the deskew range

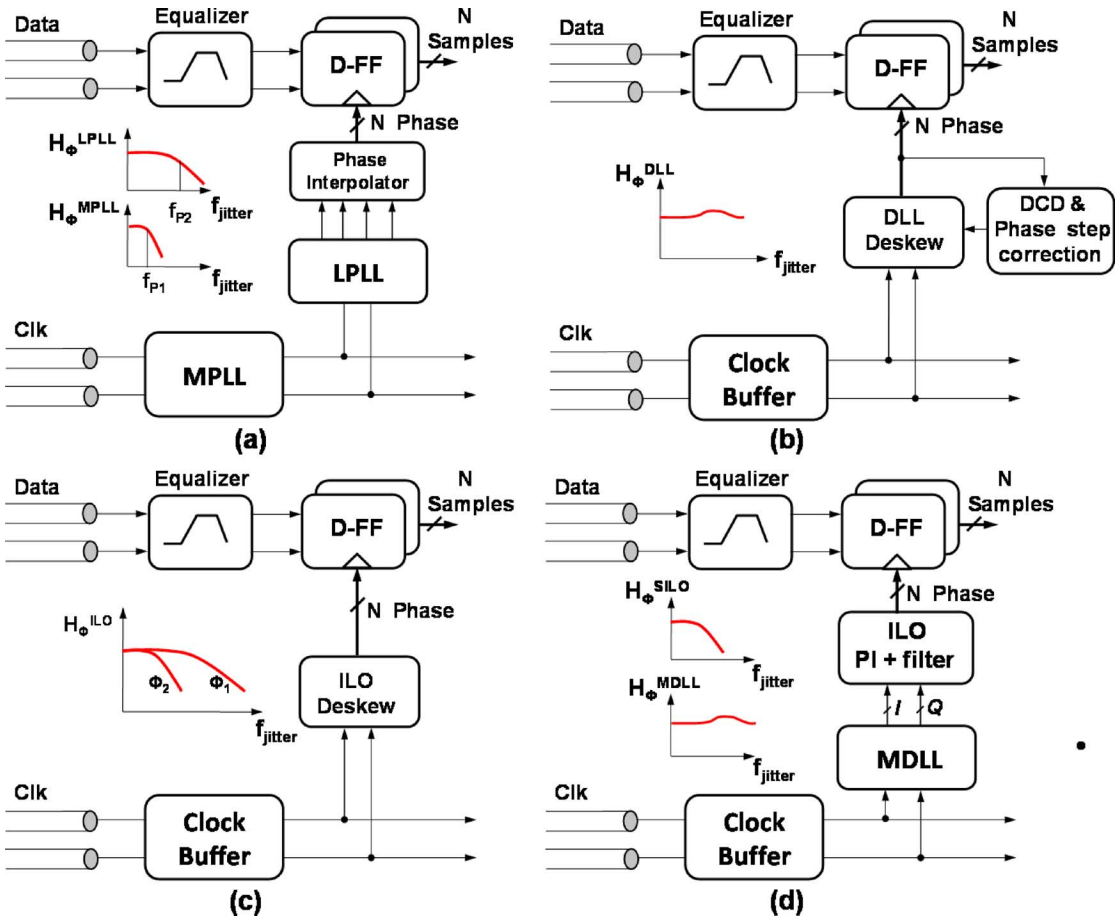


Fig. 4. Prior-art in clock-forwarded receiver architectures: (a) PLL-PLL in [8]; (b) DLL only in [6]; (c) ILO only in [10]; (d) MDLL-ILO in [12].

is $\pm 90°$. Maximum JTB is achieved for 0° phase deskew setting, e.g., ($\Delta\omega \approx 0$) and minimum JTB at 90° phase deskew. Consequently, JTB varies over several hundred MHz over different deskew settings. In addition, clock multiplication is not performed in [10], [11].

Clock multiplication can be provided using a multiplying DLL (MDLL). Unlike PLLs, MDLLs do not suffer from jitter accumulation since the phase error is reset to zero by each available reference edge. However, duty cycle distortion is not filtered by MDLLs due to their all-pass jitter transfer characteristics. In [12], an ILO is then used to interpolate between the coarse MDLL skew settings and filter out high-frequency periodic jitter generated in the MDLL, as shown in Fig. 4(d). Note that ILO is functionally equivalent to a first-order PLL where the input phase noise is low-pass filtered and the VCO's self-generated phase noise is high-pass filtered. Both the high-pass and the low-pass transfer functions have the same cut-off frequency. A high cutoff frequency is desirable to filter the VCO's phase noise and to track more correlated jitter. On the other hand, a lower cutoff frequency is more effective at filtering high-frequency periodic jitter. Since only one jitter filter appears in the clock path, its conflicting requirements lead to a sub-optimal design choice. Moreover, compared to a DLL-only or ILO-only solution, the MDLL-ILO architecture consumes more power.

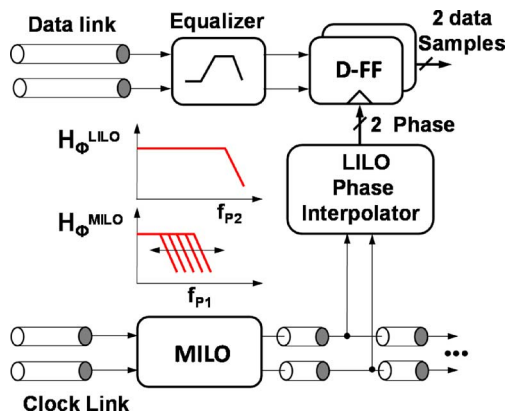


Fig. 5. Proposed clock forwarded receiver architecture.

IV. Proposed Architecture

In this work, a combination of two ILOs is used, as shown in Fig. 5: a shared ILO provides clock multiplication and optimal jitter tracking bandwidth, and a local per lane ILO provides clock deskewing to retime the data.

A. CMU and Clock Distribution

An open drain CML buffer can be used as a clock transmitter to send the clock signal over the channel and through an on-die

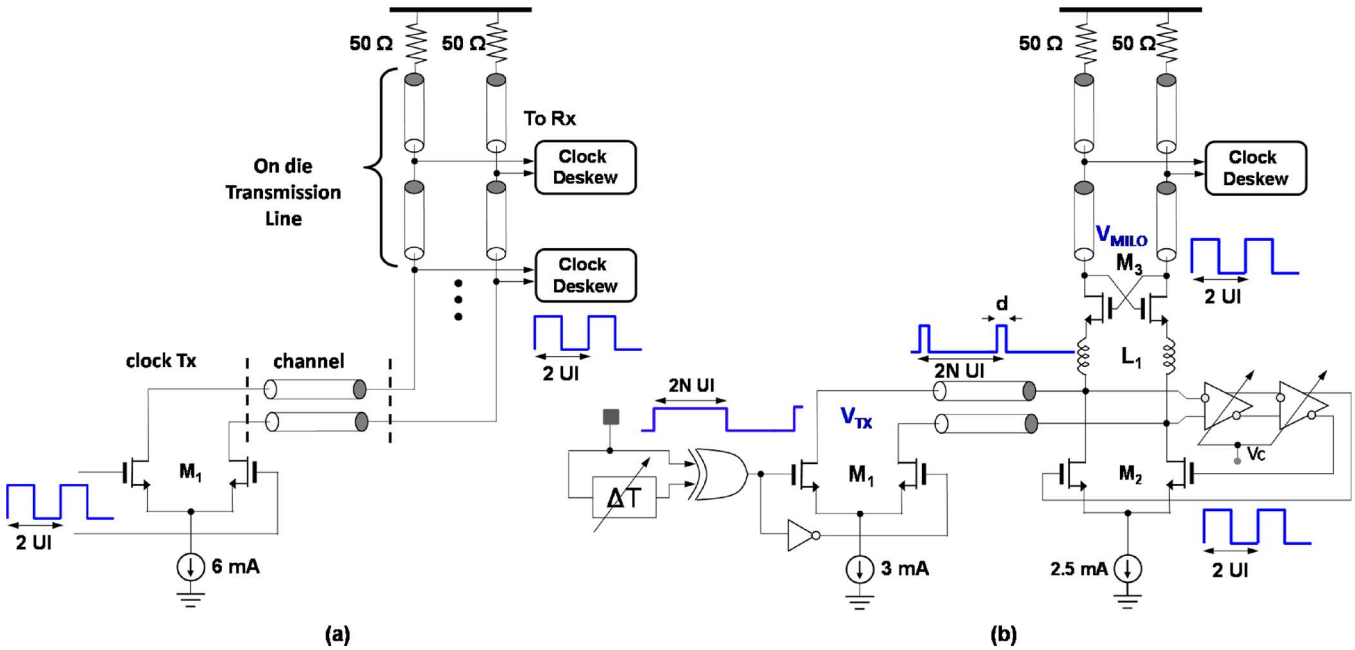


Fig. 6. (a) Passive clock distribution in a conventional clock forwarded link. (b) The proposed ILO based clock distribution. The width of M_1 and M_2 are 60 μm and the width of M_3 is 30 μm.

transmission line at the receiver side, as shown in Fig. 6(a). The CML buffer needs at least 6 mA current to achieve $>$ 200 mV swing throughout the receiver. This clock distribution provides a good compromise between power consumption, latency and supply noise rejection.

The proposed architecture, shown in Fig. 6(b), makes a simple modification to this existing clock distribution approach – a phase filter is introduced in the form of an injection locked ring VCO. A three-stage ring oscillator is used as the clock multiplying ILO (MILO) where two stages of CML inverters are tunable delay elements. The third stage in the MILO ring is used as a clock buffer as well as providing the additional gain and delay required sustain the oscillations. The total power of the clock transmitter is redistributed between the open-drain clock transmitter and the MILO buffer to achieve the same swing as before. The VCO is tunable from 1.7 to 4.5 GHz to provide a half-rate clock for 4 Gb/s to 8 Gb/s operation. Compared to existing clock distribution schemes, this ILO based clock distribution does not add any additional latency in the clock path. The only added power consumption in the proposed architecture are the two tunable delay elements.

Inductor L_1 resonates out part of the pad capacitance and device capacitance of M_2 and M_3 forming a low Q ($Q \approx$ 2 to 2.5) LC filter. Transistors M_3 serve as a cross-coupled common-gate clock buffer distributing the clock signal across 1 mm of on-die transmission line to the local injection locked oscillator (LILO). Estimated loss of this on-die transmission line is $\approx$ 1.25 dB/mm. Including the low Q LC filter, the clock distribution network suffers 3.5 to 4 dB loss at 4 GHz. Although narrow band resonant clocking can provide very low power clock distribution [8], in this work to support several data rates, a relatively low Q filter with broadband transmission line is preferred for clock distribution [4]. In addition supply induced jitter is minimized by using a CML style clock buffer M_1 and M_2. The main disadvantage of this approach is the higher loss due to lower Q and the relative power inefficiency of the CML buffer as a clock driver. Due to the limited silicon area in the implemented protoype, the MILO clock buffer is driving only one LILO and a 1 mm on-die transmission line. However, simulation results show that the existing MILO can drive up to four LILO with more than 400 mVp-p differential swing. It is well known that injection locked oscillators are functionally equivalent to a first-order PLL [13]. Thus their jitter transfer function can be written as

$$\mathrm{JTF}_{\mathrm{INPUT}}(\omega_{\mathrm{jitter}}) = \frac{1}{1 + \dfrac{j\omega_{\mathrm{jitter}}}{\omega_P}} \tag{5}$$

where the pole of the jitter transfer function can be written as

$$\omega_P = \sqrt{\frac{K^2}{A_{\mathrm{VCO}}^2} - \Delta\omega^2}. \tag{6}$$

Here, K is the injection strength, A_{VCO} captures VCO topology dependency [14] and $\Delta\omega$ is the frequency difference between free running ILO frequency ω_0 and the injected frequency ω_{inj}. To accommodate clock multiplication, a subrate clock can be forwarded and the MILO tuned to lock to one of its harmonics. For example, an ILO can provide 5$\times$ clock multiplication when locked to the 5th harmonic of the injected signal as shown in Fig. 7(a). However, injection strength K in this case is determined by the amplitude of the 5th harmonic which is only a small fraction of the fundamental tone. This can be better understood by taking the Fourier transform of the pulse train as illustrated in Fig. 8. For a given pulse amplitude h and duty cycle of 50%, the amplitude of the nth harmonic is

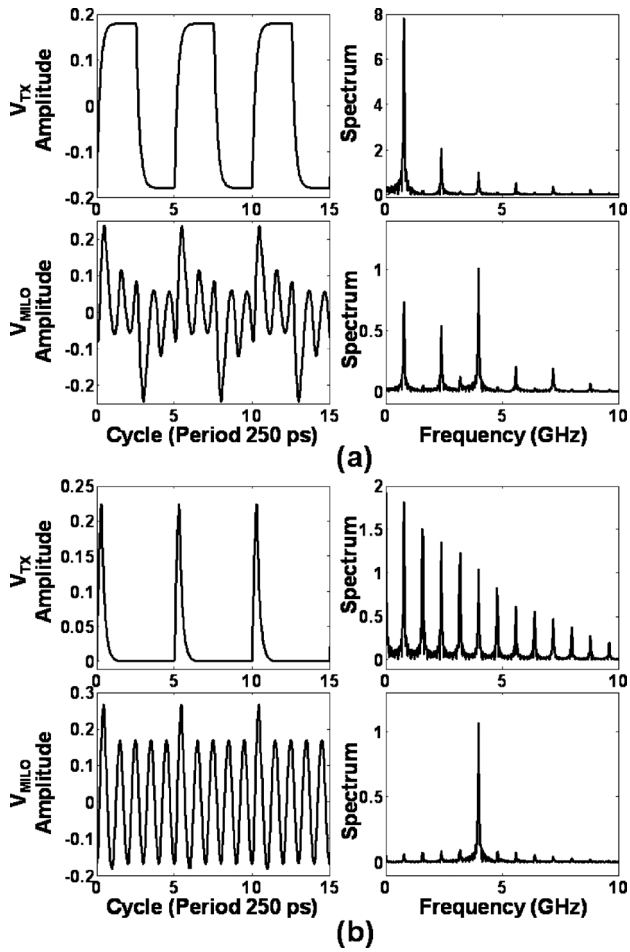


Fig. 7. The simulated transmitted signal and MILO output in the time and frequency domains: (a) NRZ; (b) pulse signal. The spectra are normalized to the power of the extracted tone at 4 GHz.

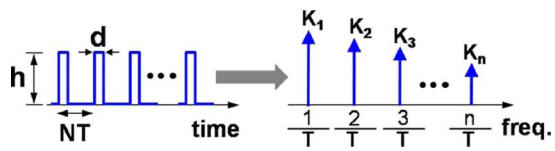


Fig. 8. Pulse train in time and frequency domain.

$2h/(n\pi)$. Hence, to obtain sufficient locking range, a lower frequency pulse train requires a larger injection amplitude h. Note that although in theory the ratio between the Nth harmonic amplitude and the fundamental is $1/N$, this ratio is reduced due to bandwidth limitations of the circuit. For example, this ratio of approximately 1/7.5 is observed for the fifth harmonic in the spectrum plot of Fig. 7(a). As a result generally if injected with a sub rate (quarter-rate or lower) clock, significant amplitude distortion and reference spurs appear at the MILO output.

This problem is ameliorated in this work by injecting a pulse train as shown in Fig. 7(b). The effective injection strength of the nth harmonic of the pulse train can be obtained using the Fourier series:

$$K_n = \frac{2h}{n\pi} \sin \frac{n\pi d}{T}. \tag{7}$$

This is illustrated in Fig. 8. For the fundamental frequency, $n = 1$, the injection strength is

$$K_1 = \frac{2h}{\pi} \sin \frac{d\pi}{T}. \tag{8}$$

Similarly, for a $1/N$ sub-rate forwarded clock,

$$K_N = \frac{2h}{N\pi} \sin \frac{d\pi}{T}. \tag{9}$$

Assuming a 10% duty cycle, the ratio between the fifth harmonic and the fundamental is approximately 3/5. Compared to injection of a subrate 50% duty-cycle clock, pulse injection provides a 3× improvement in signal to distortion ratio. Unlike NRZ signals, pulse trains effect the MILO output only at their transitions. As a result, amplitude distortion and frequency spurs are significantly reduced. Pulse trains are generated using a delay and XOR gate integrated into the clock transmitter of this prototype link [Fig. 6(b)]. Pulse width d is controlled by the delay element, ΔT. Pseudo differential CMOS inverters are used as a pre-driver of the clock transmitter. The additional delay introduced by the extra inverter in the single ended to differential converter is a small fraction of the clock period, hence negligible. Simulation results of this approach are shown in Fig. 7(b).

Using expression (9) for effective injection strength, the jitter tracking bandwidth can be written as a function of pulse repetition rate, N, and pulse duty cycle, d/NT:

$$\omega_P \approx \frac{2h}{A_{\mathrm{VCO}} N\pi} \sin \frac{d\pi}{T}. \tag{10}$$

Here, we assume negligible frequency offset ($\Delta\omega \approx 0$). This can be ensured in two ways. In the first approach, a frequency calibration loop with a replica VCO can be used for frequency tracking as reported in [15]. Since the PLL with replica VCO is always running, supply and temperature variations are tracked and compensated by the feedback loop. The only downside is the added power and area penalty of the extra PLL. In the second approach, the frequency of the free running ILO can be calibrated during start up and the control voltage set such that $f_{\mathrm{inj}} = f_{\mathrm{osc}}$. In this work we used the second approach to reduce power and complexity.

The MILO JTB is set by the effective injection strength which is controlled by changing the duty cycle, d, and pulse repetition rate, N (Fig. 9). The effect of pulse repetition rate upon tracking bandwidth is illustrated in the phase step response. The phase step responses are generated by shifting the rising edge of the input pulse train. As expected, with $\mathrm{N} = 1$ the MILO phase is updated more frequently than with $\mathrm{N} = 8$ resulting in a higher tracking bandwidth for $\mathrm{N} = 1$ than $\mathrm{N} = 8$. Tracking bandwidth can also be adjusted by adjusting the duty cycle as shown in Fig. 10, thereby providing continuous adjustment of the JTB from 25 MHz to 300 MHz. The shared clock circuitry consumes more power than any other block in the link to ensure that even when set to a low JTB, a low phase noise clock is distributed to the LILOs.

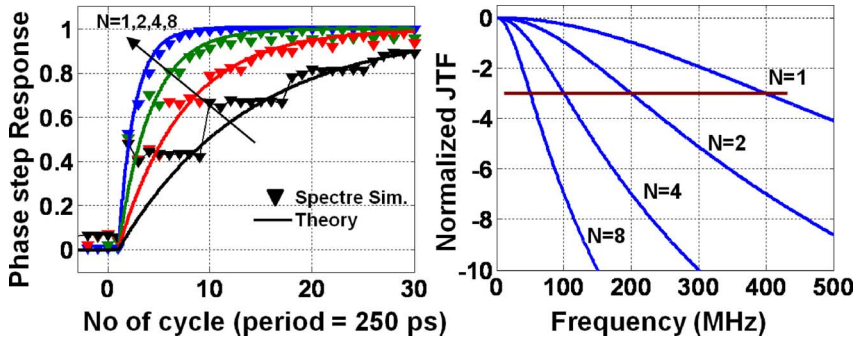


Fig. 9. The phase step response and the corresponding jitter transfer functions for different pulse repetition rates $\mathrm{N} = 1,2,4,8$.

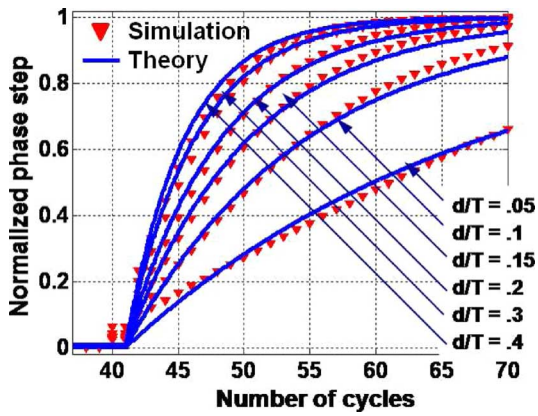


Fig. 10. Phase step response for different duty cycles.

B. Phase Interpolation With Injection Locking

The MILO output is distributed by a passive clock distribution network. This low-power clock distribution technique provides better supply rejection and lower latency than a buffered approach at the cost of smaller signal swing. Each lane of the receiver requires a deskew circuit to provide precise phase alignment, amplification and high-frequency jitter (DCD) filtering of the clock as well as generating all phases required by the sampling flip-flops. Two existing approaches to ILO-based clock deskew are shown in Fig. 11(a)-(b).

In [12] both 0 and 90 degree clock phases are combined together to inject symmetrically in every stage of the the ring oscillator [Fig. 11(a)]. By adjusting the relative strength of the injection phase, interpolation can be obtained. This method provides complete $0^\circ \longrightarrow 360^\circ$ phase interpolation and JTB remains relatively constant over the phase interpolation range. But distributing quadrature phases (0°, 90°, 180°, 270°) throughout the die consumes significant power. Alternatively, in [10] only differential phases (0°, 180°) are injected to a single point in the VCO [Fig. 11(b)]. As a result power consumption in the clock distribution network in halved. Phase interpolation is then obtained by detuning the free running ILO frequency. However, the JTF is a strong function of the frequency offset and hence phase deskew. As a result to obtain phase shifts greater than $\pm 45^\circ$, jitter tracking bandwidth significantly drops. In addition, the phase deskew range is smaller than the previous approach.

To overcome the above limitations, the proposed architecture is shown in Fig. 11(c) combining the benefits of the above architectures. Instead of combining four phases, we inject the differential clock into the ring at two points with adjustable polarity and three possible injection strengths to select between eight coarse deskew settings, as shown in Fig. 12. Interpolation between these coarse settings is done by slightly detuning the LILO's free-running frequency. The resulting phase shift, θ_{ss}, and the corresponding frequency offset, $\Delta\omega$, are related by the following expression:

$$\theta_{ss} = \sin^{-1}\left(\frac{A_{\mathrm{VCO}}}{K}\Delta\omega\right). \tag{11}$$

For the maximum required phase shift $\pm 23^\circ$, the largest required frequency offset is: $\Delta\omega_{\max} \approx K/2A_{\mathrm{VCO}} \approx 2\pi \times 500\,\mathrm{MHz}$. This translates to a minimum jitter tracking bandwidth in (6), $\omega_{P\,\min} \approx \sqrt{3/4}\times K/A_{\mathrm{VCO}} \approx 2\pi\times 750\,\mathrm{MHz}$. The LILO's measured tracking bandwidth exceeds 600 MHz so that the overall JTB of the clock path is determined by the MILO, independent of the phase deskew setting. The LILO is designed to have wide tuning range (2 GHz to 5.5 GHz) to accommodate a wide range of data rates with some additional margin for fine phase interpolation.

C. Phase Noise Filtering

The jitter transfer of the shared CMU and local phase interpolator is shown in Fig. 13. The phase noise of the CMU output can be written as

$$S_{\mathrm{CMU}}(\omega_{\mathrm{jitter}}) = \frac{\omega_{P1}^2 S_{\mathrm{ref}}(\omega_{\mathrm{jitter}}) + \omega_{\mathrm{jitter}}^2 S_{\mathrm{MILO}}(\omega_{\mathrm{jitter}})}{\omega_{P1}^2 + \omega_{\mathrm{jitter}}^2} \tag{12}$$

where S_{ref} is the jitter spectrum of the supplied reference clock, S_{MILO} is the jitter spectrum of the MILO when free-running (with no injection), and ω_{P1} is the jitter tracking bandwidth of the MILO, which depends upon the injected duty cycle and pulse repetition rate. This CMU output is then filtered by the per lane LILO

$$S_{\mathrm{out}}(\omega_{\mathrm{jitter}}) = \frac{\omega_{P2}^2 S_{\mathrm{CMU}}(\omega_{\mathrm{jitter}}) + \omega_{\mathrm{jitter}}^2 S_{\mathrm{LILO}}(\omega_{\mathrm{jitter}})}{\omega_{P2}^2 + \omega_{\mathrm{jitter}}^2} \tag{13}$$

where S_{LILO} is the free-running jitter spectrum of the LILO, and ω_{P2} is the tracking bandwidth of the LILO. This architecture provides several advantages. Phase tracking of the two loops can be set independently by appropriately choosing ω_{P1} and ω_{P2}. Here, ω_{P1} is chosen to optimize correlated jitter tracking,

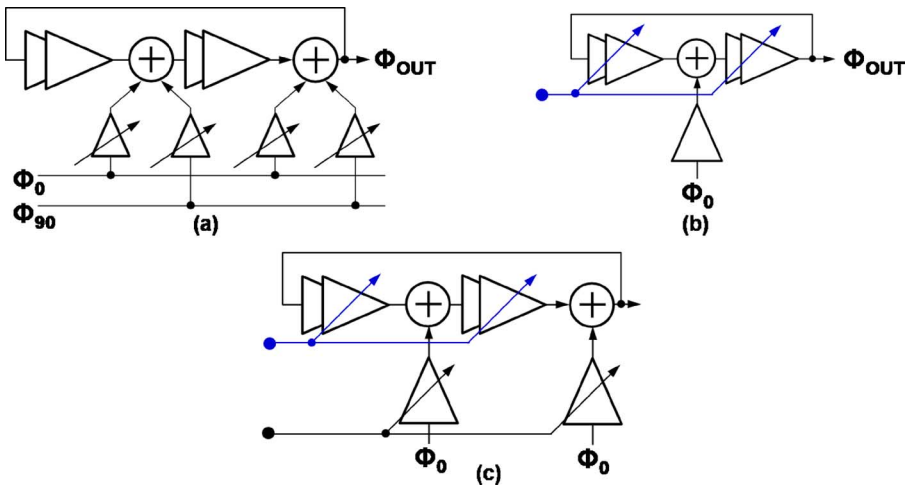


Fig. 11. ILO based phase interpolator: (a) as in [12], (b) as in [9], (c) this work.

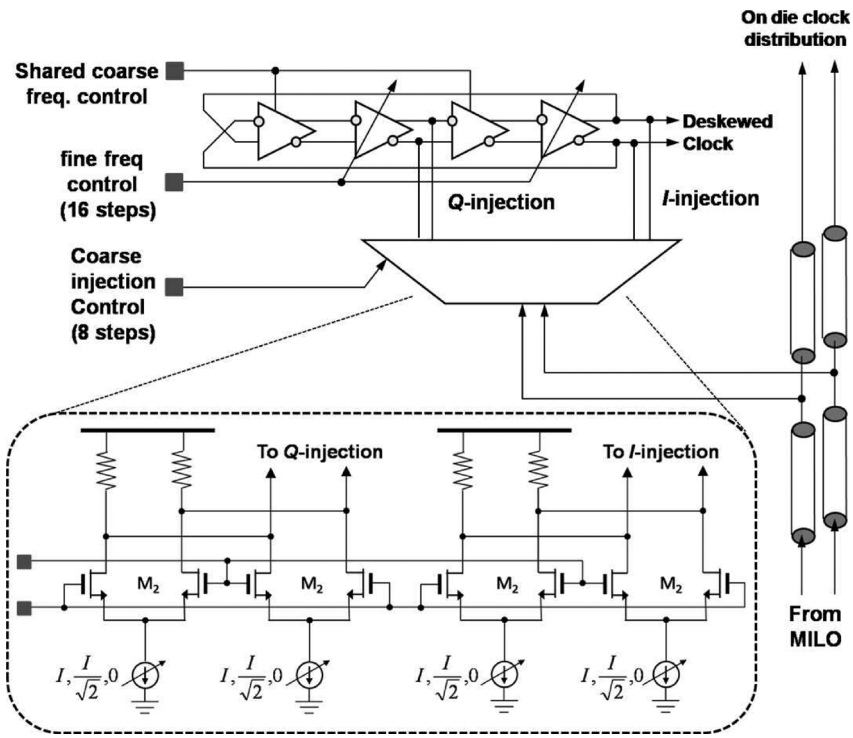


Fig. 12. The implemented ILO based phase interpolator. The width of M_2 is 20 μm.

and $\omega_{P2} \gg \omega_{P1}$ provides some additional filtering of DCD and other very high-frequency (uncorrelated) jitter. In Section II it was shown that the optimum tracking bandwidth for the clock path varies from 25 MHz to 400 MHz. From (12) and (13) and Figs. 13–14, note that in the range $\omega_{P1} < \omega_{\text{jitter}} < \omega_{P2}$, S_{MILO} is the dominant contributor to output phase noise, and its noise is not correlated with that of the data. This can be a wide range of frequencies, especially when latency mismatch between clock and data is high requiring $\omega_{P1} \ll \omega_{P2}$. Therefore it is critical to design the MILO with low phase noise. As a result, the MILO consumes more power than any other ILO. Fortunately, the MILO's power is amortized over all receiver lanes hence does not translate to a significant power penalty. On the other hand $\omega_{P2} \approx 2\pi \times 700$ MHz filters out S_{LILO} up to very high frequencies, so very little of S_{LILO} appears in the recovered clock jitter spectrum, S_{out}. As a result, the per lane LILO can be designed with low power improving receiver power efficiency. Another advantage of high tracking bandwidth in the LILO is that when used in a burst mode application, each receivers lane can wake up very quickly. Very high-frequency jitter due to DCD and reference spurs is still attenuated by both ILOs. CML delay stages are used in both the MILO and LILO providing good supply noise immunity.

V. Implementation and Experimental Results

Each receiver comprises an equalizer to compensate high-frequency channel and package losses followed by demultiplexer. The equalizer improves eye opening and reduces uncorrelated pattern dependent jitter which will not be tracked by the forwarded clock path. In a relatively low loss channel, a simple

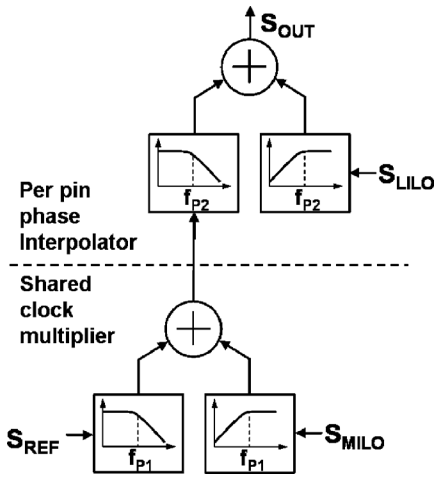


Fig. 13. Phase noise transfer model for the two cascaded ILOs.

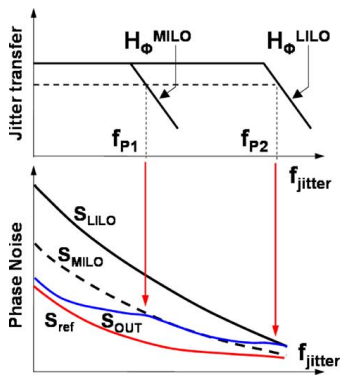


Fig. 14. Phase transfer and resultant Phase noise for the two cascaded ILOs.

passive equalizer can provide this functionality with excellent power efficiency.

A. Passive Equalizer

A passive equalizer in the form of C-R high-pass filter is used to equalize FR4 traces. The implemented equalizer circuit is shown in Fig. 15. Its transfer function is

$$\frac{V_{\text{in}}}{V_{\text{rec}}} = \frac{R_2 + sCR_1R_2}{(R_1 + R_2) + sCR_1R_2}. \tag{14}$$

The DC gain of the equalizer is $R_2/(R_1 + R_2)$ and the high-frequency gain is 1. Thus the equalizer provides a boost of $1 + R_1/R_2$. The zero and pole of the equalizer can be written as $\omega_z = 1/CR_1$ and $\omega_p = 1/(C(R_1 \| R_2))$. In reality, the location of the pole is at a slightly lower frequency due to the input capacitance. The choice of equalizer parameters R_1, R_2, C is driven by tradeoffs between input termination, input time constant and maximum boost. Small values of R_1 and R_2 degrade input matching and require larger values of C (hence area) to keep the zero at the same frequency. Larger value resistors increase the input time constant. Fortunately, a 1:2 demultiplexer introduces less loading than a higher order demultiplexer. To adjust the location of the zero for different data rates and channel characteristics, the capacitance C can be adjusted using switches as shown in Fig. 15(a). Channel responses (20-inch long FR-4 trace) with and without the

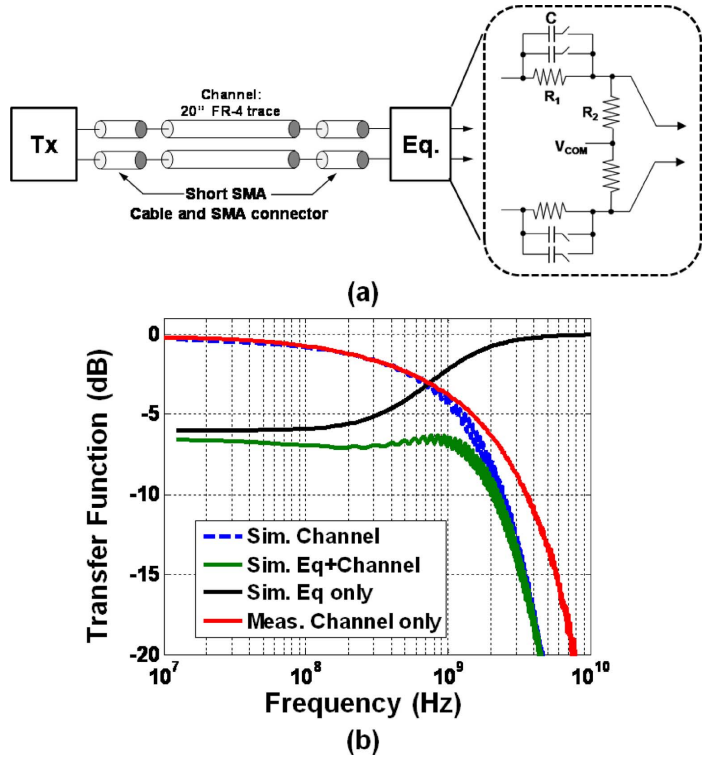


Fig. 15. (a) The implemented passive equalizer and experimental setup. (b) The equalizer frequency response for (20-inch) FR4 PCB trace with and without equalizer.

equalizer are shown in Fig. 15(b). Note that the packaged prototypes are connected through SMA cables, connectors and the channel. No drilled vias, daughter card or other board edge connectors are used in the experimental setup. The channel measurements do not include the pad capacitance or I/O device capacitances. However, these were included in the simulated channel. Approximately 1 pF additional capacitance is used in simulation, accounting for the discrepancy between the simulated and measured channel responses in Fig. 15(b). Although the passive equalizer achieves excellent power efficiency their usefulness is limited to well behaved channels with less than 10 dB loss at the Nyquist rate and relatively slow roll-off (20 dB/decade or lower). Since the passive equalizer does not provide signal amplification, in lossy channels it can degrade receiver sensitivity. For channels with faster roll-off, it is difficult to invert the channel response with a single zero and a single pole.

B. Experimental Results

The 4–7.4 Gb/s 65 nm CMOS receiver prototype is tested in a QFN package and operates from a 1 V supply. A die photo and power break down are shown in Fig. 16.

The shared clock circuitry consumes 8 mW, the LILO phase interpolator consumes 4.4 mW and the samplers consume 2.4 mW. Excluding shared clock power, each receiver consumes 6.8 mW which equals 0.92 pJ/bit at 7.4 Gb/s. A demonstration of 16× clock multiplication is shown in Fig. 17. The delay-XOR combination generates a pulse train that in frequency domain is a series of impulses spaced 250 MHz apart [Fig. 17(a)-(b)]. The MILO locks to the tone at 4 GHz and suppresses the

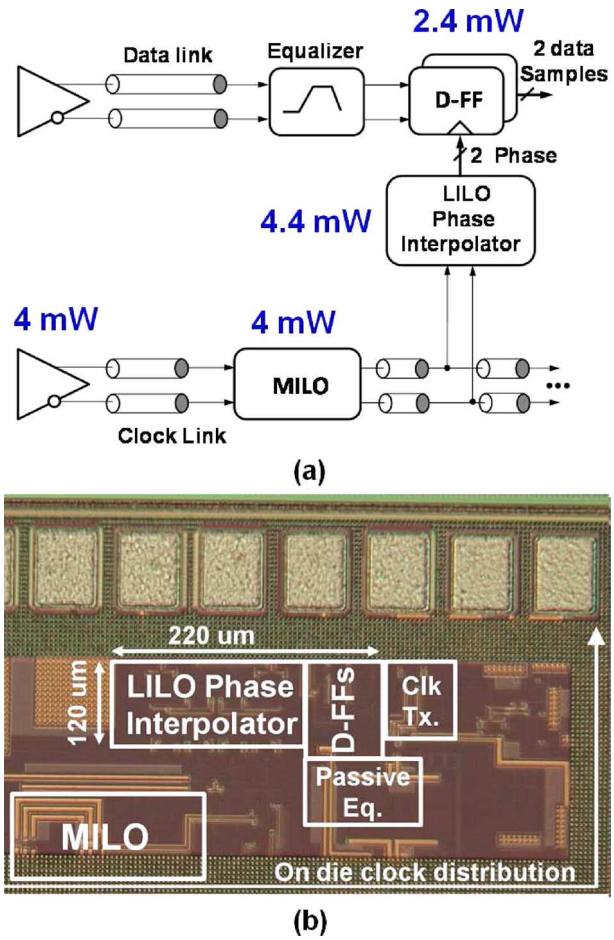


Fig. 16. Implemented protoype. (a) Block diagram with power breakdown. (b) Die photo of the prototype in 65 nm CMOS.

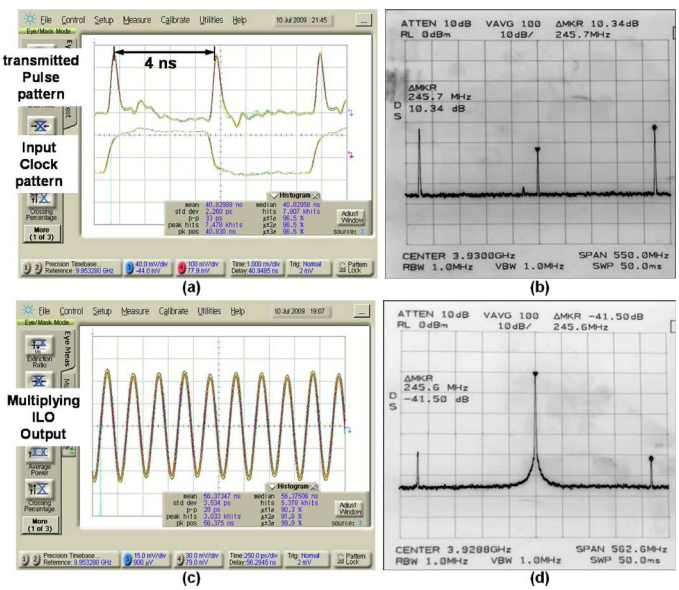


Fig. 17. Verification of 16× clock multiplication. (a) Pulse train in time domain. (b) Pulse train in frequency domain. (c) Recovered clock in time domain (d) Recovered clock in frequency domain.

other tones due both to the inherent phase noise filtering of the MILO and the incorporated low Q passive resonator. The recovered 4 GHz clock in time and frequency domain is shown

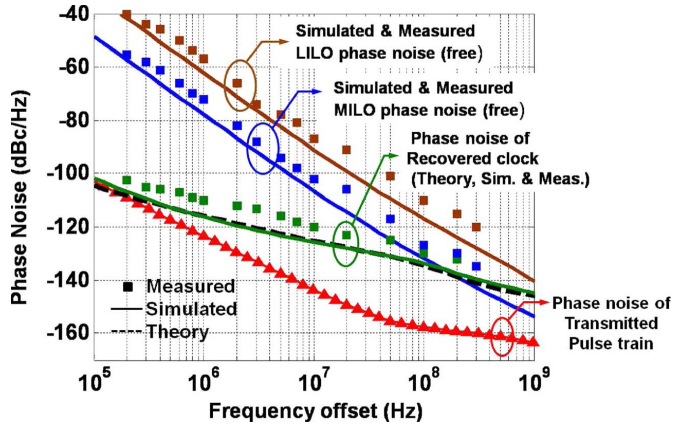
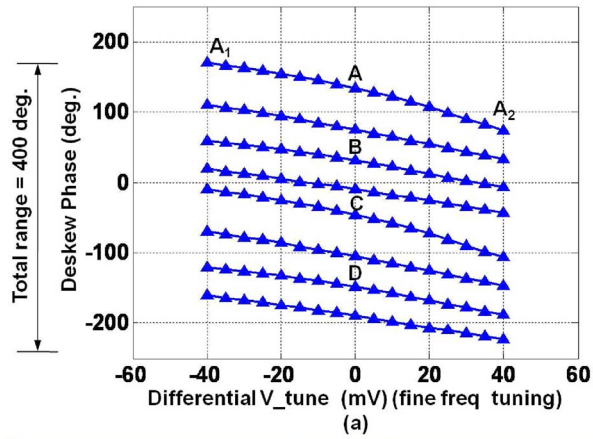


Fig. 18. Phase noise of the free running ILOs and of the recovered clock.



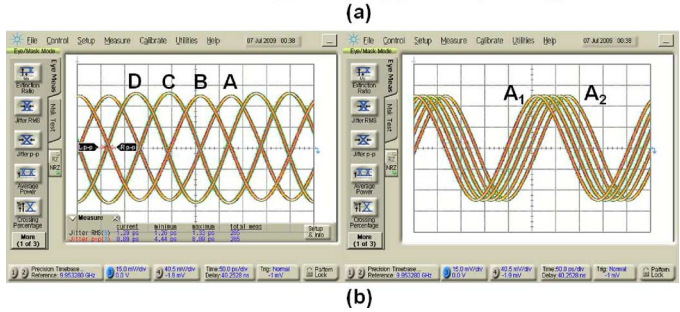


Fig. 19. (a) Measured deskew with coarse and fine control. (b) Four specific coarse and fine phase settings.

in Fig. 17(c)-(d) and has −41 dBc reference spurs. Those spurs add < 1.5 ps to the total jitter using the formula for spurious tonal jitter (spurious jitter $= 10^{\mathrm{spur}/20}/\pi f_{\mathrm{osc}}$).

The phase noise of the clock reference, MILO and LILO are shown in Fig. 18. As explained in Section IV(C), the MILO's contribution to recovered phase noise is dominant over a wide range of frequency offsets. Thus the MILO was designed to have at least 15 dB lower phase noise than the LILO at frequency offsets of 25 MHz to 1 GHz. The LILO phase noise is filtered out of the final recovered clock up to 1 GHz. Coarse and fine phase interpolation with the ILO is demonstrated in Fig. 19. Coarse selections are set by the different injection locations, polarities, and strengths while fine interpolation curves are generated by detuning the free running frequency of the LILO. The fine tuning curves each exceed the spacing between neighboring coarse tuning settings by at least 60%. The linearity of this phase

TABLE I
COMPARISON WITH STATE-OF-THE-ART LOW POWER CLOCK FORWARDED RECEIVERS

	[8]	[6]	[10]	[11]	[12]	(This Work)
Architecture	PLL-PI	DLL	ILO	ILO	MDLL-ILO	MILO-ILO
Data Rate	6.25Gb/s	5Gb/s	27Gb/s	7.2Gb/s	8Gb/s	7.4Gb/s
Clock Multiplication Ratio	16x	1x	1x	1x	1x,2x,4x,5x, 8x,10x	1x,2x,4x, 8x,16x
Clock frequency	$\frac{1}{2}$-rate	$\frac{1}{4}$-rate	$\frac{1}{2}$- rate	$\frac{1}{4}$-rate	$\frac{1}{4}$-rate	$\frac{1}{2}$- rate
Ref. Spur	——	——	———	———	-32 dBc	-41.5 dBc
Clock Jitter (ps rms)	——	——	———	1.4-3ps	1.8-6.6ps	1.4ps (N=1) 4ps(N=16)
Jitter Tracking BW	——	All pass	100MHz-750MHz (varies w/ phase)	30MHz - 100MHz	——	25MHz-300MHz
Jitter tolerance (UIpp)	——	0.72UI @ 15MHz 0.3UI @ 100MHz	——-	——-	——	1.5UI@ 200MHz
Technology	90nm	0.13um	45nm	90nm	0.13um	65nm
Rx Power	8.2mW	32.25mW	43mW	4.3mW	33mW	6.8mW
CMU Power	3.6mW	——	——	——	0	4mW (+4mW tx)
FOM (pJ/bit)	1.31	6.45	1.6	0.6	4.125	0.92

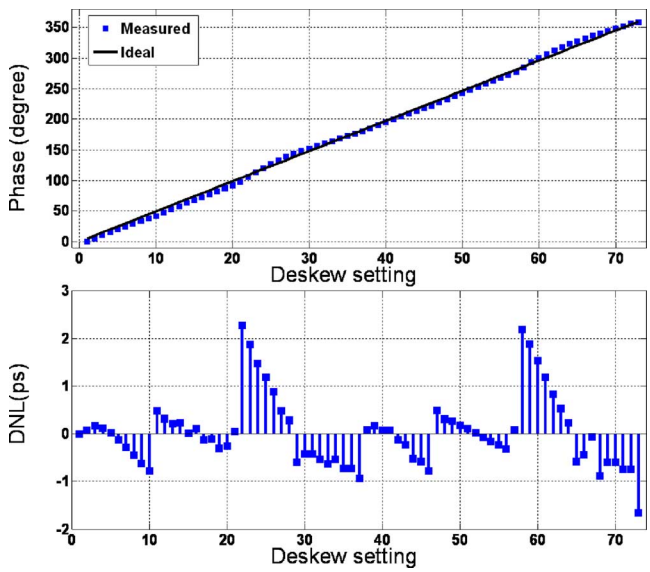


Fig. 20. Output phase as a function of deskew setting and the corresponding DNL at a clock frequency of 3.7 GHz (7.4 Gb/s data rate).

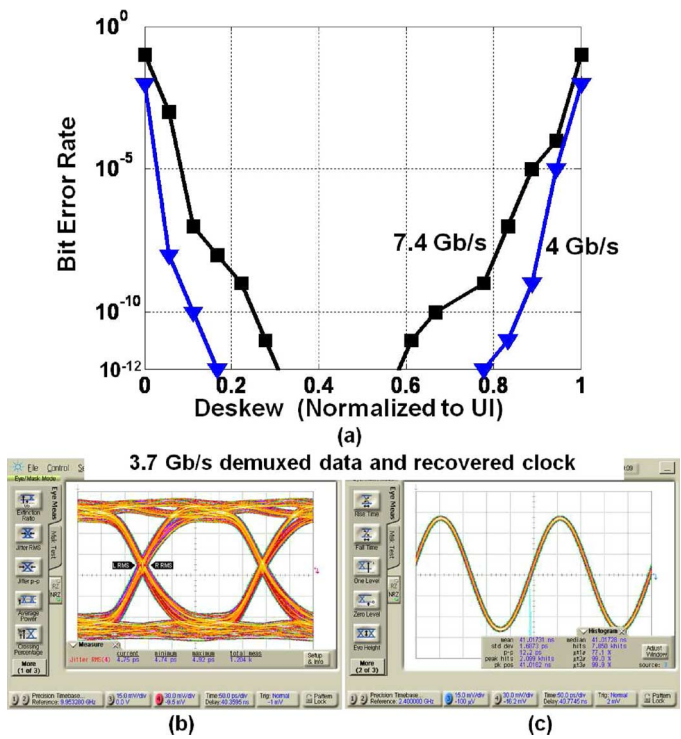


Fig. 21. (a) BER as a function of phase deskew at 4 Gb/s and 7.4 Gb/s over 10" and 5" FR4 traces, respectively. The BER is measured with a $2^{31}-1$ pattern. (b) Half-rate recovered data. (c) Half-rate recovered clock.

interpolator is shown in Fig. 20. The DNL is better than 3 ps, with discontinuities of <3 ps observed when switching from one coarse tuning setting to another. The BER of the receiver for a $2^{31}-1$ pattern is shown in Fig. 21 as a function of deskew setting over 5-inch FR4 interconnect. Note that due to the finite resolution in skew settings and residual jitter in the recovered clock the BER at 4 Gb/s does not quite reach $10^{-0.3} = 0.5$. Jitter tolerance is tested at 7.4 Gb/s and plotted in Fig. 22. Jitter transfer is captured for three pulse repition rates, N = 1, 2, and 4. In all cases measured rtesults are in good agreement with the theory. In addition, latency mismatch between the clock and data paths was kept low (< 2 UI) in the experimental setup. As a result jitter tolerance improved with increasing tracking bandwidth. The BER is less than 10^{-12} in the presence of 1.5 UI (peak-to-peak) sinusoidal PJ at 200 MHz. This is in addition to 0.45 UI (peak-to-peak) deterministic jitter which was observed due to the simple and, hence, imperfect passive equalization.

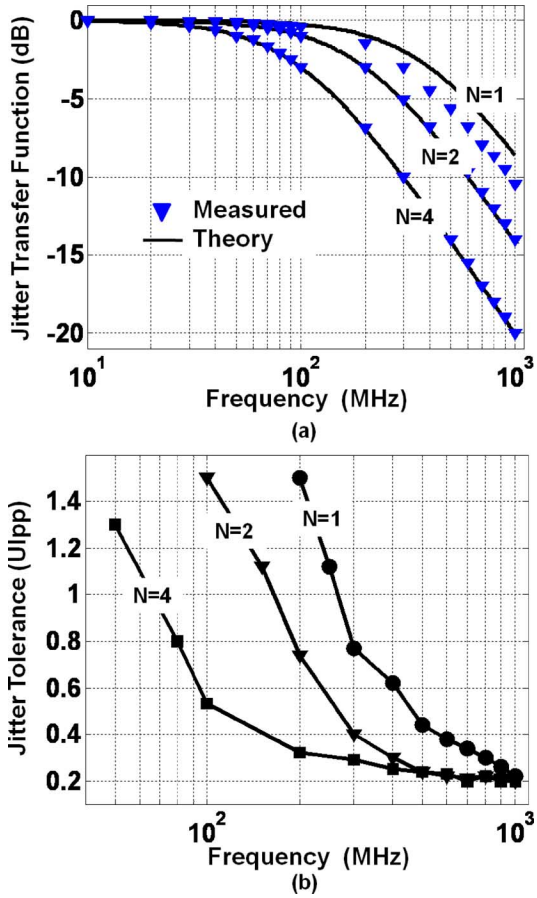


Fig. 22. (a) The clock path's measured jitter transfer. (b) The corresponding link jitter tolerance.

VI. Conclusion

The proposed architecture is compared with state-of-the-art receivers in Table I. All the architecture previously described in Section III are considered in this comparison. Notice that ILO based solutions have achieved significantly better power efficiency than PLL or DLL based solutions. However, in other works they did not provide clock multiplication. The proposed solution combines the functionality of PLL- or DLL-based solutions without sacrificing the excellent power- and area-efficiency offered by injection locking. The high-frequency jitter tolerance achieved (1.5 UI at 200 MHz) is comparable to oversampling CDRs, a significant improvement over previous low-power clock forwarded receivers.

References

[1] N. Kurd *et al.*, "Next generation Intel® Core™ micro-architecture (Nehalem) clocking," *IEEE J. Solid-State Circuits*, pp. 1121–1129, Apr. 2009.

[2] R. Schmitt, L. Hai, C. Madden, and C. Yuan, "Investigating the impact of supply noise on the jitter in gigabit I/O interfaces," *IEEE Electrical Performance of Electronic Packaging*, pp. 189–192, Oct. 2007.

[3] E. Yeung and M. Horowitz, "A 2.4 Gb/s/pin simultaneous bidirectional parallel link with per-pin skew compensation," *IEEE J. Solid-State Circuits*, vol. 35, no. 11, pp. 1628–1699, Nov. 2000.

[4] E. Prete, D. Scheideler, and A. Sanders, "A 100 mW 9.6 Gb/s transceiver in 90 nm CMOS for next-generation memory interfaces," in *IEEE Int. Solid-State Circuits Conf. Dig.*, 2006, pp. 90–91.

[5] K. Lee *et al.*, "A jitter tolerant 4.5 Gb/s CMOS interconnect for digital display," in *IEEE Int. Solid-State Circuits Conf. Dig.*, 1998, pp. 90–91.

[6] A. Agrawal, P. Hanumolu, and G. Wei, "A 8 × 5 Gb/s source-synchronous receiver with clock generator phase mismatch correction," presented at the 2008 IEEE Custom Integrated Circuits Conf., San Jose, CA, Sep. 2008.

[7] R. Kho *et al.*, "A 75 nm 7 Gb/s/pin 1 Gb GDDR5 graphics memory device with bandwidth improvement techniques," *IEEE J. Solid-State Circuits*, vol. 45, no. 1, pp. 120–133, Jan. 2010.

[8] J. Poulton *et al.*, "A 14-mw 6.25-Gb/s transceiver in 90-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2745–2757, Dec. 2007.

[9] F. O'Mahony *et al.*, "A low-jitter PLL and repeaterless network for a 20Gb/s link," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, 2006, p. 29.

[10] F. O'mahony *et al.*, "A 27 Gb/s forwarded clock I/O receiver using an injection-locked LC-DCO in 45 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig.*, 2008, p. 452.

[11] K. Hu *et al.*, "A 0.6 mW/Gbps, 6.4–8.0 Gbps serial link receiver using local injection-locked ring oscillators in 90 nm CMOS," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, 2009, pp. 46–47.

[12] H. Ng *et al.*, "A second-order semidigital clock recovery circuit based on injection locking," *IEEE J. Solid-State Circuits*, vol. 38, no. 12, pp. 2101–2110, Dec. 2003.

[13] R. Adler, "A study of locking phenomena in oscillators," *Proc. IRE*, vol. 33, pp. 351–357, Jun. 1946.

[14] M. Hossain and A. Chan Carusone, "CMOS oscillators for clock distribution and injection-locked deskew," *IEEE J. Solid-State Circuits*, vol. 44, no. 8, pp. 2138–2153, Aug. 2009.

[15] F. O'Mahoney, B. Casper, M. Mansuri, and M. Hossain, "A programmable phase rotator based on time-modulated injection-locking," in *IEEE Symp. VLSI Circuits Dig. Tech. Papers*, 2010, pp. 45–46.

Masum Hossain (M'11) received the B.Sc. degree in electrical engineering from the Bangladesh University of Engineering and Technology, Bangladesh, and the M.Sc. degree from Queen's University, Canada, in 2002 and 2005, respectively. During his M.Sc. work, he worked on K-band wireless receiver in CMOS. Since 2005, he has been working towards the Ph.D. degree in electrical engineering at the University of Toronto, Canada.

From September 2007 to January 2008, he was with Intel Circuit Research Lab (CRL) as a graduate intern. From 2008 to 2010 he was with Gennum Corp. in the Analog and Mixed Signal division. Currently, he is a senior member of technical staff at Rambus Lab. His research interests include mixed-signal circuits for high-speed chip-to-chip communications, low-power VCO, phase interpolator and clock recovery techniques.

Mr. Hossain won the best student paper award at the 2008 IEEE Custom Integrated Circuits (CICC) Conference.

Anthony Chan Carusone (S'96–M'02–SM'08) received the Ph.D. degree from the University of Toronto, Canada, in 2002.

Since 2001 he has been with the Department of Electrical and Computer Engineering at the University of Toronto where he is currently an Associate Professor. In 2008 he was a visiting researcher at the University of Pavia, Italy, and later at the Circuits Research Lab of Intel Corp., Hillsboro, Oregon. He is also an occasional consultant to industry in the areas of integrated circuit design, clocking, and digital communication.

Prof. Chan Carusone was a co-author of the best student papers at both the 2007 and 2008 Custom Integrated Circuits Conferences and the best paper at the 2005 Compound Semiconductor Integrated Circuits Symposium. He is an appointed member of the Administrative Committee of the IEEE Solid-State Circuits Society and sits on the Executive Committee of the Circuits and Systems Society's Board of Governors. He served on technical program committee for the IEEE Custom Integrated Circuits Conference, and is currently a member of the technical program committee for the VLSI Circuits Symposium. He has served as a guest editor for the IEEE Transactions on Circuits and Systems I: Regular Papers and was on the editorial board of the IEEE Transactions on Circuits and Systems II: Express Briefs from 2006 until 2009 when he was Editor-in-Chief. He is currently an associate editor of the IEEE Journal of Solid-State Circuits.