A 10Gb/s Data-Dependent Jitter Equalizer

James Buckwalter and Ali Hajimiri

California Institute of Technology
Pasadena, CA 91125 USA

Abstract

An equalization circuit is presented that reduces data-dependent jitter by aligning data transition deviations. This paper presents an analytic solution to data-dependent jitter and demonstrates its impact on the phase noise of the recovered clock. A data-dependent jitter equalizer is presented that compensates the impairment of the channel and lowers the phase noise of the recovered clock. The circuit is implemented in a SiGe BiCMOS process and operates at 10Gb/s. It suppresses phase noise resulting from data-dependent jitter by 10 dB.

Introduction

Signal integrity issues such as timing jitter are at the forefront of high-speed digital design. Electronic circuit speeds are overwhelming the legacy channels that traditionally could be treated as ideal. Henceforth, the channel behavior must be compensated appropriately to reach the highest information capacity. Noise considerations dictate the choice of equalization technique.

Jitter is deviations in the timing of received data bits compared to a reference such as the transmitter. Data jitter reduces the horizontal opening of the data eye. Furthermore, the sampling clock is recovered from the edges of received data. Therefore, the data jitter deviations translate to phase noise in the recovered clock and consequently sampling uncertainty in the data eye. This uncertainty reduces the receiver bit error rate (BER) performance.

BER requirements compel limiting the jitter from the standpoint of decision errors and the performance of the clock and data recovery (CDR). Alternatively, managing jitter can loosen the restrictions on the jitter transfer and, hence, the bandwidth of the clock recovery, reducing acquisition time of the CDR.

Timing jitter is composed of random jitter (RJ) and deterministic jitter (DJ). RJ results from random voltage noise occurring during data transitions and phase noise of the transmitter [1]. DJ is timing deviations of the data transitions correlated to non-ideal behavior of system blocks such as limited bandwidth, signal reflection, duty cycle distortion, or power supply noise [1]. Depending on the source, DJ is classified into subcategories. Data-dependent jitter (DDJ) is a prominent form of DJ caused by the random nature of previously transmitted data symbols. Studies on DDJ properties are presented in [2]-[4].

In this paper, the behavior of DDJ for general pulse responses in optical channels and transmission lines is summarized. Next, we describe the impairment of DDJ on the recovered clock. We describe qualitatively the relationship of data jitter to other sources of jitter in the CDR. Recent work on jitter in phase-locked loop circuits developed closed loop parameter adaptation for minimizing the total clock jitter [5].

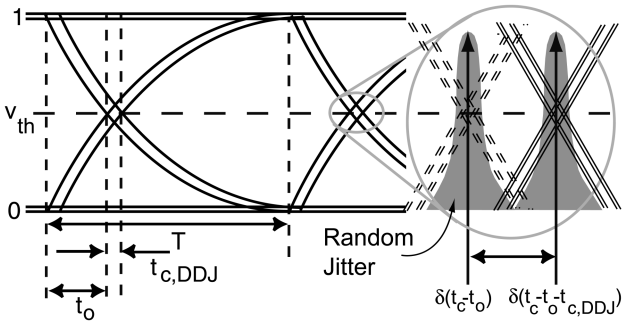

Fig. 1 The data-dependent jitter apparent in the data eye and the resulting jitter

This work offers instead a circuit technique to minimize data-dependent jitter. The results of the analysis of data-dependent jitter suggest leveraging the intrinsically deterministic nature of data-dependent jitter to dynamically adjust the receiver response. Though the data is generated stochastically, the effect on the transition time can be determined once the data value has been decided in the receiver. The advantages of this technique are discussed.

Finally, a SiGe implementation of the DDJ equalizer is demonstrated that works to 10.5 Gb/s. We verify the operation of the equalizer by introducing a jittered input waveform and measuring the phase noise of the recovered clock. Then we compensate the waveform and measure the phase noise of the recovered clock. Additionally, the timing jitter of the recovered clock is measured and improvement is demonstrated.

Analysis of Data-Dependent Jitter

The response of a causal system with finite bandwidth to a pseudorandom bit sequence (PRBS) is not only determined by the current bit but also the previously transmitted bits. Effectively, the system retains memory of the previous bits. At each transition, the sequence of previous bits shifts the output amplitude and changes the relative time the signal crosses a decision threshold. This effect is illustrated in Fig. 1. This timing deviation depends on the particular data sequence.

The response of the channel as well as the transmitter and receiver determines the behavior of DDJ. In the absence of noise, a received data signal, $r(t)$, is

$$r(t) = \sum_{n=-\infty}^{0} a_n g(t - nT). \quad (1)$$

Here, a_n is the data symbol and $g(t)$ is the received pulse shape for a bit period of *T*. This pulse shape is determined by an appropriate modulation scheme and the response of the channel. Now we determine the threshold crossing time, t_c, for arbitrary values of previous bits.

$$v_{th} = r(t_c) = \sum_{n=-\infty}^{0} a_n g(t_c - nT) \quad (2)$$

where v_{th} is the decision threshold level. In general, (2) cannot be solved implicitly for t_c in a closed form solution. However, linearization techniques such as the Taylor series expansion approximate the DDJ [4]. A first-order Taylor series is sufficient for smooth pulse responses.

$$g(t - nT) = g(t_o - nT) + (t - t_o)g'(t_o - nT) \quad (3)$$

where $g'(t)$ is the derivative of *g* with respect to time. The estimated threshold crossing time, t_o, is derived from the time a single step crosses v_{th}. Substituting (3) into (2), the DDJ is

$$t_{c,DDJ} = t_c - t_o = \frac{v_{th} - \sum_{n=-\infty}^{0} a_n g(t_o - nT)}{\sum_{n=-\infty}^{0} a_n g'(t_o - nT)} . \quad (4)$$

The form of $g(t)$ determines the DDJ properties. Notably, the denominator contains the slope and the numerator contains the value of the pulse shape. Equation (4) can be simplified if we consider only three bits. For 001 and 110 sequences, (4) is zero since, by definition, $g(t_o) = v_{th}$. For the 101 and 010 sequences,

$$t_{c,DDJ} = \frac{v_{th} - g(t_o + T)}{g'(t_o + T)} . \quad (5)$$

Since, these data sequences are equally probable and we expect the data transitions to jump between the t_o and $t_o + t_{c,DDJ}$. NRZ data is generated stochastically and DDJ is a probability density function (PDF) with discrete values of t_c. For the 001 and 101 sequences, the first-order DDJ PDF consists of a double dirac function as modeled in [1].

$$pdf_{DDJ}(t_c) = \frac{1}{2}[\delta(t_c - t_o) + \delta(t_c - t_o - t_{c,DDJ})] \quad (6)$$

Since RJ and DDJ are independent, the total jitter PDF is the convolution of the RJ and DDJ PDFs.

$$pdf_{TotalJitter} = pdf_{RJ} \otimes pdf_{DDJ} \quad (7)$$

RJ is typically a Gaussian distribution and the DDJ peaks are described in (6). Therefore, the RJ is mapped onto each DDJ peak, as illustrated in Fig. 1. The root-mean-square (rms) magnitude of the total jitter increases with the separation between the delta functions described in (6). Therefore, this is an important characteristic of the deterministic and, consequently, total jitter [1]-[4]. Using a model for the pulse response, an expression for DDJ is found in (5) and the jitter peaks of (6) are predicted.

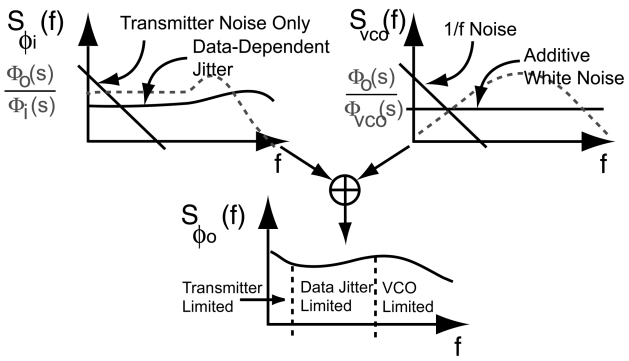

Fig. 2 The different noise sources superimposed on the transfer functions to demonstrate the impact on the recovered clock phase noise.

DDJ Impact on Clock and Data Recovery

Clock and data recovery is often accomplished with a phase-locked loop (PLL) that detects the phase of the received data. DDJ degrades the performance of the PLL by changing the phase from cycle to cycle. An estimate of the jitter spectral density (JSD) is given by the phase variance of the crossing times.

$$\sigma_\phi^2 = E[(\phi_c - \phi_{c,m})^2] = (2\pi/T)^2 E[(t_c - t_{c,m})^2] \quad (8)$$

where $E[\ ]$ is the expectation operation and the data crossing phase, ϕ_c, is normalized by the mean value, $\phi_{c,m}$. This is reasonable because the PLL will attempt to lock to the mean of the phases. Therefore, the DDJ JSD of the received data is

$$S_{\phi_i}(f) = \sigma_\phi^2 = (\pi t_{c,DDJ}/T)^2 . \quad (9)$$

The JSD is a white noise source where the noise level is dictated by the ratio of the separation of the DDJ peaks and the bit period.

The importance of DDJ on PLL performance depends upon the contribution of other jitter sources. PLLs suffer from additional noise sources, as described in [6] and demonstrated in Fig. 2. The external noise sources include timing deviations due to data jitter and phase noise of the transmitter. Since data are multiplexed into a single serial bit stream, noise of the transmitter clock directly affects the data transition time. The external jitter sources appear as clock jitter according to $\Phi_o(s)/\Phi_i(s)$, which is a low-pass response. Therefore, they dominate at low frequencies.

The voltage controlled oscillator (VCO) contributes to internal noise. Oftentimes, a white and 1/*f* noise source is added to the VCO tuning port to model oscillator noise. The transfer function, $\Phi_o(s)/\Phi_{VCO}(s)$, of the internal noise sources to the output phase is high-pass and the low-frequency noise of the local oscillator is less critical than the 1/*f* noise of the transmitter oscillator.

Therefore, the phase noise of the recovered clock is composed of three regions. At very low frequencies, the noise of the transmitter clock (often 1/*f* noise) is dominant. At higher frequencies, the JSD overwhelms the 1/*f* noise and dominates the phase noise. However, near the bandwidth of the loop filter these terms are attenuated and the internal noise of the VCO determines the phase noise of the recovered clock. These regions are summarized in Fig. 2.

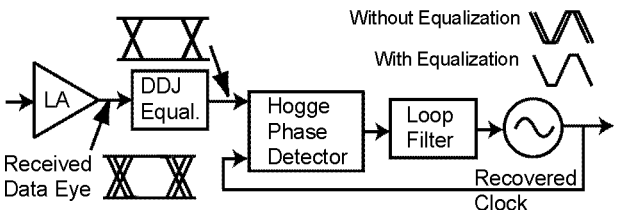

Fig. 3 The block diagram for a CDR circuit with jitter equalization.

DDJ Equalization

From the analysis of DDJ, the PDF was described by two dirac delta functions in (6). The separation between these peaks was related to the pulse response and the previous bits described in (5). Therefore, the DDJ is equalized by deterministically adjusting the receiver path based on the state of the previous bits. This overlays the peaks of the DDJ PDF and reduces the total jitter spectral density of the input data.

Since removing DDJ improves the BER of the receiver by increasing the timing margins as well as decreasing the phase noise of the recovered clock, the optimum location for the DDJ equalizer is before the CDR and decision device as shown in Fig. 3. This manages the DDJ accumulated through the transmitter, channel, and receiver. Additionally, it is noteworthy that DDJ equalization can be implemented as a pre-emphasis technique. Accordingly, the DDJ peaks at the transmitter are reversed such that at the receiver they overlap.

One algorithm that accomplishes this alignment is sketched in Fig. 4. The equalizer is nonlinear and is not disturbed by the limiting amplifier (LA). During each bit period, the previous bits are compared, in this case with a XOR gate, to determine whether a transition has occurred between the last two bits. If a transition has occurred, the algorithm adjusts the delay time and, consequently, the transition time according to a equalizer tuning voltage. Otherwise the delay path is held at a nominal value.

Theoretically, this equalization scheme should minimize the jitter spectral density to the point that the transmitter clock phase noise, the random data jitter, and the VCO phase noise dominate over the entire region.

More importantly, this equalizer is superior to an inverse channel filter because it does not amplify high-frequency noise. As long as the jitter generation of the time delay switching stage is low, the dynamic operation of the equalizer is different than a linear equalizer with a static high-pass characteristic. The DDJ equalizer contributes little additional jitter because during the data transition the equalizer has reached a new fixed delay value.

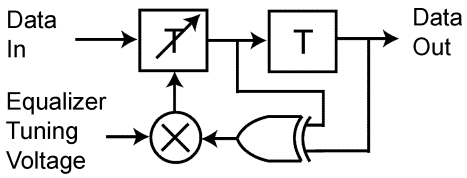

Fig. 4 The block diagram for a DDJ equalizer.

Circuit Implementation

The circuit is implemented in a SiGe BiCMOS process. A chip microphotograph is provided in Fig. 5. The chip area measures 1.2mm by 1mm including the loop filter and pads.

The circuit consists of the described DDJ equalizer, a PLL, and 50 Ohm output drivers. The PLL is designed with a modified Hogge phase detector (PD) [7][8]. This phase detector is robust to variations of the pattern density, transition density, and duty-cycle. Additionally, the Hogge PD comprises cascaded DFFs. The output of each flip-flop is a bit period shifted version of the received signal. The DDJ equalizer taps the input of the Hogge phase detector and the first DFF as demonstrated in Fig. 6. Accordingly, the circuit overhead is reduced. Since the delays depend on the DFF driven by the recovered clock, the equalization only operates correctly when the PLL is locked.

The delay stage is based on a current-starved differential pair suggested in [9]. Each path is driven by the transition-detection multiplier implemented with an XOR logic gate. This approach was preferred to multiplexing a delayed and undelayed version of the signal because the implemented delay stage requires lower power consumption and avoids glitches in the 2:1 multiplexer.

The operation of the DDJ equalizer requires that the current is fully steered within one bit period. NMOS transistors provide smooth transition of the current between the differential pairs. Degenerated bipolar devices can be used alternatively. The amount of variation between the two delay values is controlled by the equalizer tuning voltage. This device acts as a current by-pass to the transition-detection multiplier.

The phase detector drives a differential charge pump (CP). The loop filter is designed with a bandwidth of about 1 MHz. The loop filter generates a differential control voltage for the complementary cross-coupled oscillator. The tuning range of the oscillator is 9 to 11.5GHz to provide robustness to process variations. The output of the oscillator is buffered and drives the phase detector and a 50 Ohm buffer.

The circuit consumes 70mA from a -3.5 V supply. The DDJ equalizer draws 10mA of this current.

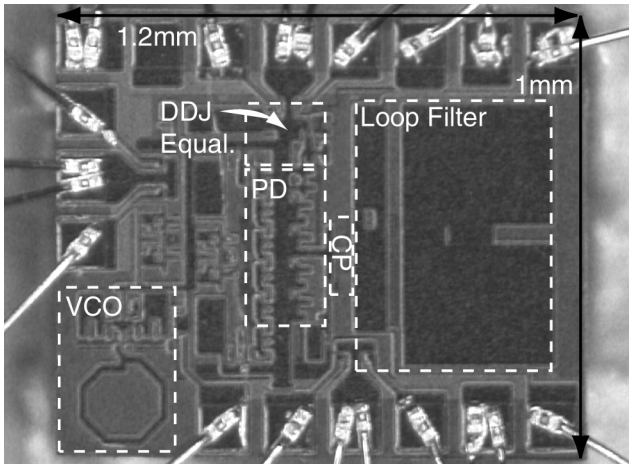

Fig. 5 Chip microphotograph of the CDR and DDJ equalizer.

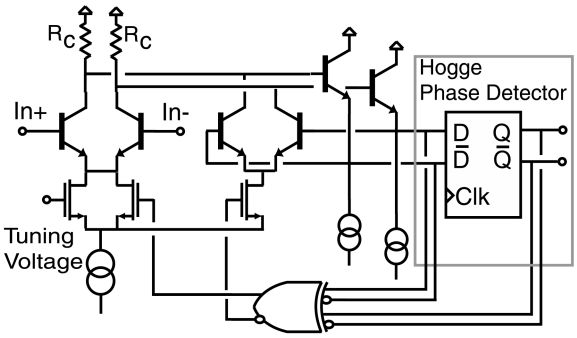

Fig. 6 Schematic for fully differential DDJ equalizer built into Hogge phase detector.

Results

The performance was measured using a single-ended signal from a 10Gb/s Anritsu MP1763C PRBS generator. A short transmission line of four inches of FR-4 board was introduced to the data path to generate additional data-dependent jitter. The measured DDJ of the microstrip line is about 6ps. The chip was mounted and wirebonded to a Rogers 6010 duroid board. The additional behavior of the wirebonds influence the exact DDJ value.

The phase noise of the oscillator was measured with and without equalization. Originally, the equalizer tuning voltage was set so that the variable delay is zero. The PLL was locked to a 2^7-1 PRBS sequence and the phase noise of the locked VCO was measured. The uncompensated phase noise is depicted in Fig. 7. The prominent regions include the data jitter spectral density and the VCO phase noise. Clearly, in the data jitter limited region the phase noise of the oscillator is about -88 dBc/Hz.

Next, the equalizer tuning voltage was increased and the phase noise of the VCO decreased. Fig. 7 demonstrates the improvement in the phase noise. The phase noise at 100kHz offset drops by 12dB with equalization and by 8dB at 1MHz offset. The new phase noise floor indicates that the DDJ is less than 1ps and is limited instead by random jitter.

Additionally, the measured clock jitter is demonstrated in Fig. 8. The jitter drops from an rms value of 2.6ps before equalization to 1.9ps at the appropriate equalizer tuning voltage. This reflects a significant improvement in the quality of the clock in Fig. 8.

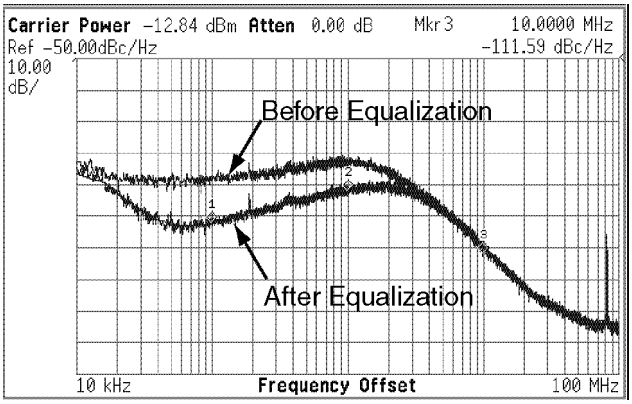

Fig. 7 Phase noise of the recovered clock with and without equalization.

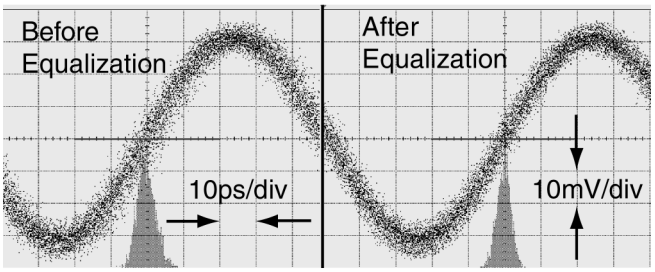

Fig. 8 Clock jitter with and without equalization.

Conclusions

An analysis of data-dependent jitter in general LTI systems is presented. The analysis demonstrates that data-dependent jitter can be compensated by varying the response of the clock and data recovery circuit according to whether a transition has occurred before the previous bit. An equalizer implementing this algorithm is proposed. The equalizer is implemented with a 10Gb/s clock and data recovery circuit and the performance of the equalizer is measured by the reduction in the phase noise of the PLL. Over 10dB improvement is measured around the offset most affected by data jitter. Furthermore, the clock jitter is reduced by 0.7ps. The equalization concept is general to a broad class of applications sensitive to data-dependent jitter.

Acknowledgments

The authors acknowledge Behnam Analui from the CHIC group at Caltech for helpful discussions. This work was funded through the support of the Lee Center for Advanced Networking. Additionally, the support of Rogers Corporation through the generous donation of PCB material is appreciated.

References

[1] International Committee for Information Technology Standardization (INCITS), Fibre Channel - Methodologies for Jitter and Signal Quality Specification- MJSQ, Technical Report REV 10.0, March 10, 2003.

[2] J. Buckwalter, B. Analui, and A. Hajimiri, "Predicting Data-dependent Jitter," in press.

[3] J. Buckwalter, B. Analui, and A. Hajimiri, "Data-dependent Jitter and Crosstalk-Induced Bounded Uncorrelated Jitter in Copper Interconects," in press.

[4] B. Analui, J. Buckwalter, and A. Hajimiri, "Data-dependent Jitter in Serial Communication," unpublished.

[5] M. Mansuri and C.-K. K. Yang, "Jitter Optimization Based on Phase-Locked Loop Design Parameters," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 1375-1382, Nov. 2002.

[6] A. Hajimiri, "Noise in Phase-Locked Loops," 2001 *Southwest Symposium on Mixed Signal Design*. Feb. 2001.

[7] C. R. Hogge, "A Self-Correcting Clock Recovery Circuit," *IEEE Journal of Lightwave Technology*, vol. 3, pp. 1312-1314, Dec. 1985.

[8] L. DeVito, "A Versatile Clock Recovery Architecture and Monolithic Implementation," *Monolithic Phase-Locked Loops and Clock Recovery Circuits,* B. Razavi, Ed., New York: IEEE Press, 1996.

[9] T. H. Lee and J. F. Bulzacchelli, "A 155 MHz Clock Recovery Delay- and Phase-Locked Loop," *IEEE Journal of Solid State Circuits*, vol. 27, pp.1736-1746, Dec. 1992.