A 2.56 Gbps Asynchronous Serial Transceiver with Embedded 80 Mbps Secondary Data Transmission Capability in 65nm CMOS

Xiaoran Wang, Tianwei Liu, Shita Guo, Mitchell A. Thornton, and Ping Gui
Department of Electrical Engineering, Southern Methodist University, USA
xiaoranw@smu.edu, tianweil@smu.edu, shitag@smu.edu, mitch@smu.edu, pgui@smu.edu

***Abstract*— A new asynchronous serial transceiver is proposed that is capable of transmitting and receiving a secondary data stream along with the primary data stream on a single asynchronous serial link. The proposed transceiver embeds the secondary data stream by modulating the phase of the primary data in accordance with it. The receiver recovers both the primary and secondary data simultaneously. In a standard receiver, which is not equipped with the phase demodulation capability, the secondary data appears as jitter of the primary data. The jitter caused by the secondary data still falls within the jitter budget of the transceiver, and having this much jitter would not adversely affect the functionality of the primary data recovery. The proposed system can be widely used in many data communication applications such as for transmitting a hidden signature for data authentication, or as control and/or additional data in an existing serial link. A prototype transceiver, implemented in a 65 nm CMOS process, demonstrates the proposed concept with 2.56 Gbps primary data and 80 Mbps secondary data channels.**

***Index Terms*— Asynchronous serial transceiver, clock and data recovery (CDR), secondary channel.**

I. INTRODUCTION

Asynchronous serial transceivers are ubiquitous in today's devices and are used to interface between blocks within integrated circuits, between ICs, and between packaged systems, typically using industry standards such as USB, MIPI, and PCI-e [1]. It is highly desirable to efficiently utilize all available bandwidth in asynchronous serial channels to increase throughput, improve data integrity, and enhance hardware security.

We propose a new asynchronous serial data transceiver that is capable of transmitting and receiving a secondary data stream along with the standard primary data stream on a single asynchronous serial link. Our proposed technique can be used to provide benefits in the abovementioned aspects of asynchronous serial links. For example, data throughput can be increased through the use of the additional secondary channel. Alternatively, the secondary channel may be used for error correction or detection on the primary channel, thereby enhancing its data integrity and capacity without an increase in bandwidth. The proposed architecture can also be applied in different ways to enhance hardware security, such as by using the secondary channel to carry authentication information for the simultaneously received primary data. Other enhancements may exploit the fact that the secondary data transmission can also be considered as a form of steganography or covert channel since it appears as jitter to a non-equipped transceiver [2]-[3]. Furthermore, the proposed method provides backward-compatibility with standard transceivers for the primary data channel.

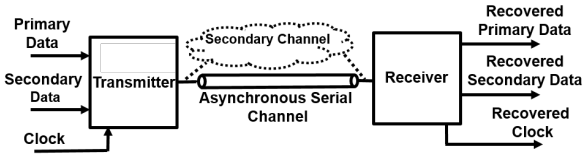

Fig. 1. Proposed asynchronous serial link with embedded secondary channel.

Most asynchronous serial protocols require a clock and data recovery (CDR) circuit at the receiver, as the timing is embedded within the data. Since the serial data stream is received in the presence of additive noise, practical systems must be designed with sufficient bandwidth to allow for reliable communications in accordance with Shannon's capacity theorem, $C = B \cdot \log_2(1 + SNR)$, where C represents the channel capacity in bit/sec, B represents the bandwidth in Hz, and SNR represents the signal-to-noise-ratio (SNR) at the receiving end.

This paper presents a transceiver that provides a cost-effective secondary data channel in the existing serial data channel, without requiring an increase in the system's bandwidth. As shown in Fig. 1, the proposed transceiver system transmits the primary and secondary data stream through a single asynchronous serial channel and simultaneously recovers both of them at the receiver.

II. TRANSCEIVER IMPLEMENTATION

The transceiver implementation described here is applicable to wireline baseband-modulated systems. Thus, the implemented transceiver embeds the secondary data into the primary data using modulation that may be considered as phase modulation (PM) since the phase of the primary data stream is conditionally delayed depending upon the secondary data values. The transceiver simultaneously recovers both the primary and the secondary data at the receiver.

A. *Transceiver Architecture*

A serial transmitter transmitting data at N bits per second typically employs a D flip-flop (DFF) triggered by a clock of frequency N Hz at the very end to synchronize every data bit before sending it out as serial data. The duration of every data bit is the same as that of one period of the clock. In our transmitter, the secondary data is used to modulate this clock to the DFF. This is implemented using a 2:1 multiplexer (MUX) whose two data inputs are the clock CLK0 and its delayed version CLK1. The MUX control input is driven by the secondary data as shown in Fig. 2. Thus, either CLK0 or CLK1 is selected for the DFF clock input by each bit of the secondary data signal to generate the modulated clock signal.

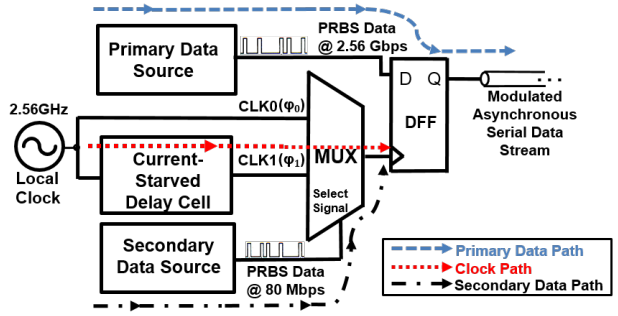

Fig. 2. Block diagram of the transmitter.

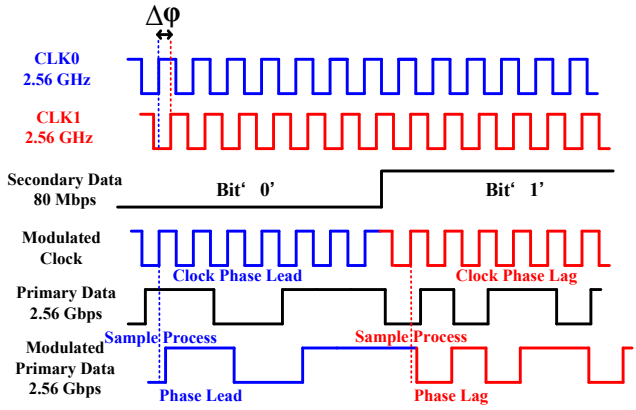

Fig. 3. Signal waveforms of the modulation scheme.

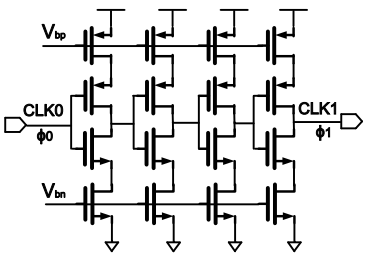

Fig. 4. Current-starved delay cell.

Fig. 3 depicts the waveforms of the clocks and data in the transmitter. CLK1 is delayed from CLK0 by Δφ. The instantaneous modulated clock is produced according to the secondary data bit. For example, when the secondary data bit is ‘0’, CLK0 will be used for the DFF and the modulated data stream is synchronized with CLK0. When the secondary data bit is ‘1’, CLK1 is used for the DFF and the modulated data stream is synchronized with CLK1. Essentially, the ‘0’ and ‘1’ bits of the secondary data are translated to phase lead and phase lag in the modulated data stream by way of the MUX serving as a binary mixer for CLK0 and CLK1.

Fig. 4 shows the current-starved delay cell that consists of four cascaded current-starved inverters. The bias voltages, V_{bp} and V_{bn}, are used to adjust the current through the top and bottom transistors thus controlling the degree of the phase shift.

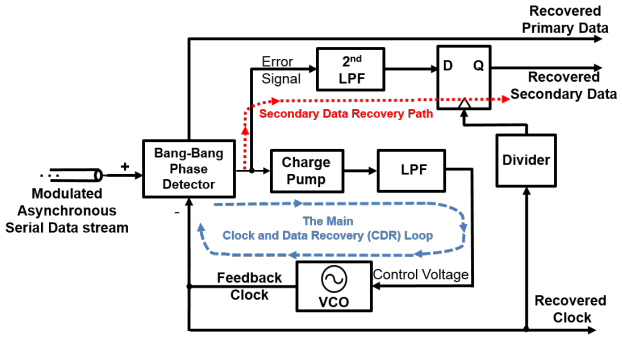

Fig. 5. Block diagram of the receiver.

Fig. 5 depicts the block diagram of the asynchronous serial receiver. It contains a main CDR circuit to recover the primary data and the clock, and a secondary data recovery path to extract the secondary data from the modulated phase of the primary data. The main CDR loop is composed of a bang-bang phase detector (BBPD), a charge pump followed by a low-pass filter (LPF), and a LC tank voltage controlled oscillator (VCO) [4]-[6].

In particular, the BBPD is shared by both the main CDR loop and the secondary data recovery path. Unlike a conventional Alexander BBPD [7], a modified BBPD is employed to detect the phase difference between the input data and feedback clock and produces either a ‘0’ or ‘1’ output as an error signal. Fig. 6 depicts the architecture and the timing diagram of the BBPD. As shown in the figure, the uniqueness of the modified BBPD is that its output is either ‘1’ to indicate a phase lag or ‘0’ to indicate a phase lead. This is different from a conventional Alexander BBPD that produces pulsed waveforms. The error signal produced by the BBPD is sent to both the charge pump in the main CDR loop and is also used in the secondary data path for demodulation of the secondary data. As the bits ‘0’ and ‘1’ of the secondary data are

embedded as phase lead and lag in the primary data stream at the transmitter, the BBPD error signal contains the demodulated phase information (lead/lag) that is used to recover the secondary data.

Some high-frequency glitches would be generated in the BBPD output, which may result from the noise coupled from the input serial stream and the VCO feedback clock. The BBPD compares each transition edge of the input serial stream with that of the VCO feedback clock to produce the error signal. When the instant random noise of the input serial stream and the VCO feedback clock is larger than the modulated phase difference Δφ, some high-frequency glitches will appear in the error signal. To fully recover the secondary data from the error signal, a second-order low-pass filter (2nd LPF) is employed in the secondary data recovery path to first filter out the high-frequency glitches in the BBPD output. By choosing an appropriate bandwidth (which will be explained in the next sub-section), the 2nd LPF can filter out the high-frequency glitches of the error signal, while maintaining the recovered secondary data. The secondary data rate is much lower than the primary data rate but they are synchronized with each other. The clock for the secondary data generation is thus a divided-down version of the faster clock in the transmitter/receiver (fast clock being the PLL output for the transmitter and the VCO output for the receiver CDR).

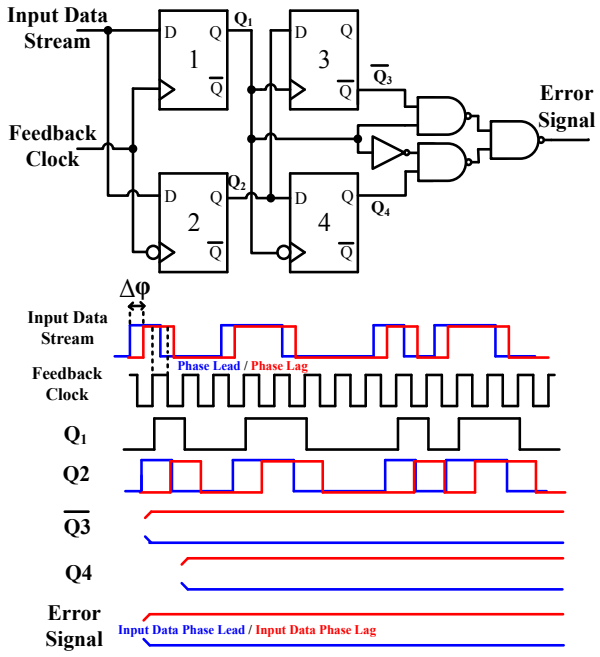

Fig. 6. The architecture and timing diagram of the BBPD.

B. System Design Parameters

In order to demodulate and recover the secondary data and to minimize the jitter in the recovered primary data, the bandwidth of the CDR, the data rate of the secondary-channel, and the bandwidth of the 2nd LPF all need to be chosen appropriately. Due to the low-pass characteristics of the CDR loop, the VCO feedback clock can track the low-frequency (in-band) phase change of the input data stream, and accordingly, the in-band phase noise remains at the recovered primary data, while the out-of-band phase noise appears at the error signal. In order to keep the secondary data information intact in the BBPD error signal, the bandwidth of the main CDR loop needs to be smaller than the low frequency components of the secondary data frequency spectrum. Meanwhile, the secondary data rate cannot be too high either. If the secondary data rate is set too large, in order to recover it successfully, the bandwidth of the 2nd LPF needs to increase accordingly. However, the increase of the 2nd LPF bandwidth may result in the failure of filtering out the high-frequency glitches from the error signal, and may degrade the BER performance of the recovered secondary data significantly. As the frequency of the unwanted glitches highly depends on the primary data rate, the secondary data rate needs to be set smaller than the primary data rate to guarantee the high-frequency glitches coupling from the noise of the primary data and the VCO feedback clock can be filtered out by the 2nd LPF. Taking the analysis above into consideration, in the initial implementation reported here, the bandwidth of the CDR is set at 1.54 MHz (2.56GHz/1667), and the secondary data rate is set as 80 Mbps. The bandwidth of the 2nd LPF is set to 40 MHz to filter out the high-frequency glitches while keeping the recovered secondary data.

To ensure the main CDR loop operates reliably, the total jitter from the transmitter and receiver must be less than 1 Unit Interval (UI), otherwise the feedback clock of the CDR would not sample the input data stream correctly, thus leading to the degradation of the BER performance of the recovered primary data. To fully recover the secondary data, the total jitter from the transmitter, the receiver, and the serial channel together needs to be less than the deterministic jitter caused by the modulated phase difference Δφ at the transmitter, otherwise the embedded secondary data information would be buried in the system's jitter. In our design, a jitter budget of <0.3 UI is allocated to the total jitter of the transceiver and the serial channel, and a jitter budget of 0.3-0.5 UI is allocated to the phase difference Δφ. Δφ is set to be about 135° in our design, which can be translated into about 0.38 UI. A ±10% PVT (process, voltage and temperature) variation of the phase difference Δφ (0.34 UI – 0.42 UI) is still within the jitter budget.

III. Measurement Results

The prototype IC for the proposed asynchronous serial transceiver was fabricated in a 65 nm CMOS process. The

die photo is shown in Fig. 7. The core circuit occupies about 0.13 mm^2, among which 0.03 mm^2 is for the transmitter, and 0.1 mm^2 is for the receiver. The power consumption of the core circuits is 11.8 mW at a supply voltage of 1.2 V. The CDR in the receiver can cover the primary data rate from 2.2 Gbps to 3.6 Gbps (54.7%). Fig. 8 shows the recovered 2.56 GHz clock signal with 31 ps total jitter, when the secondary data is embedded in the serial link. Fig. 9 shows the eye diagram of the recovered 2.56 Gbps primary data at PRBS-7, with 48 ps total jitter when the secondary data is off, and with 70 ps total jitter when the secondary data is on. Fig. 10 shows the eye diagram of the recovered 80 Mbps secondary data with 103 ps total jitter. The total jitter is measured under the targeted BER of 10^{-12}.

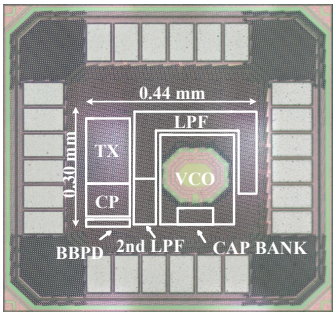

Fig. 7. Die photo of the transceiver.

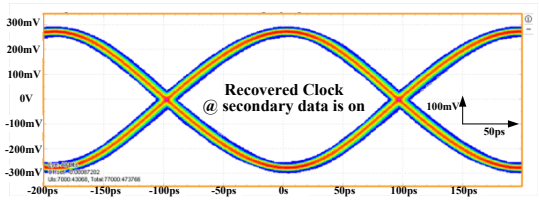

Fig. 8. The recovered clock signal.

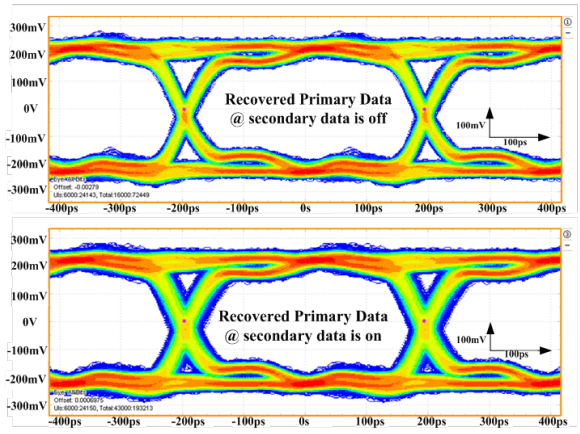

Fig. 9. Eye diagram of the recovered primary data.

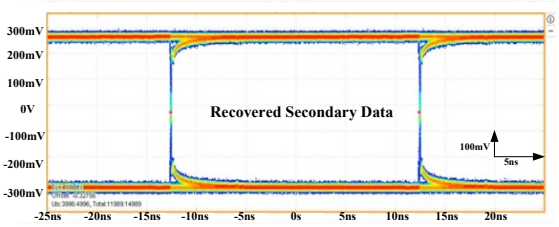

Fig. 10. Eye diagram of the recovered secondary data.

IV. Conclusion

An asynchronous serial transceiver, equipped with an additional secondary data transmission capability, was demonstrated in a 0.13 mm^2 65 nm CMOS die. The prototype transceiver allowed for both its primary and secondary data streams, at 2.56 Gbps and 80 Mbps respectively, to be recovered simultaneously with good jitter and BER performance. The proposed architecture embeds the secondary data stream in the primary data stream of a conventional asynchronous serial transceiver at a very small additional cost. This capability increases the total throughput of the system, while also offering a power-efficient solution. More importantly, the proposed serial transceiver is backward-compatible with those that are not enabled to access the secondary data, thereby allowing various unique applications.

References

[1] B. Casper and F. O'Mahony, "Clocking Analysis, Implementation and Measurement Techniques for High-Speed Data Links – A Tutorial," *IEEE Transactions on Circuits and Systems I: Regular Papers (TCAS-I),* vol. 56, no. 1, pp. 17-39, Feb. 2009.

[2] S. Zerafshan Goher, et al., "Covert channel detection: A survey based analysis," *International Conference on High Capacity Optical Networks and Enabling Technologies (HONET)*, Dec. 2012.

[3] S. Zander, G. Armitage, and P. Branch, "A Survey of Covert Channels and Countermeasures in Computer Network Protocols," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 3, October 2007, pp. 44-57.

[4] K. Fukuda, et al., "A 12.3mW 12.5Gb/s complete transceiver in 65nm CMOS," *IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC),* Mar. 2010.

[5] S. Guo, et al., "A Low-Voltage Low-Power 25 Gb/s Clock and Data Recovery with Equalizer in 65 nm CMOS," *IEEE Radio Frequency Integrated Circuits Symposium (RFIC),* pp. 307-310, 2015.

[6] T. Liu, et al., "A Temperature Compensated Triple-Path PLL with K_{VCO} Non-Linearity Desensitization Capable of Operating at 77 K," *IEEE Transactions on Circuits and Systems I: Regular Papers (TCAS-I),* vol. 64, no. 11, pp. 2835-2843, Nov. 2017

[7] C. Sanchez-Azqueta, et al., "Bang-bang phase detector model revisited," *IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2013.