Clock and Data Recovery with Adaptive Loop Gain for Spread Spectrum SerDes Applications

Ming-ta Hsieh and Gerald E. Sobelman
Department of Electrical and Computer Engineering
University of Minnesota
200 Union Street S.E., Minneapolis, MN 55455, USA

***Abstract*—A novel clock and data recovery architecture with adaptive loop gain is proposed for spread spectrum SerDes applications such as the Serial AT Attachment. The proposed design consists of a half-rate Alexander phase detector, a phase-shifting phase interpolator with a frequency differentiator and an adaptive loop gain filter. The frequency differentiator determines the clock rate difference between the referenced clock and the recovered clock. This value is then used to adjust the gain of the adaptive loop filter for better acquisition of lock with minimized jitter. The proposed design can be implemented in a digital CMOS process which reduces the design difficulty and cost. The system operation has been verified using the Cadence SpectreRF and Verilog-A simulators. The results show that the system is capable of recovering a ±5000 ppm spread spectrum data with up to a maximum of 0.5 UI of deterministic jitter.**

I. INTRODUCTION

Clock and data recovery (CDR) has played an important role in modern high-speed serial data transmission applications such as optical communications, backplane routing and chip-to-chip interconnection. The transmitted data in a communications system are often corrupted by both external and internal noise, which leads to jitter and skew in the received data. Furthermore, high-speed serial data transmission systems are often operated in an asynchronous manner between the transmitter and receiver. Therefore, the receiving end must have a CDR circuit to extract the clock signal from the received data. In addition, the received data must also be retimed and latched using the recovered clock. Reference [1] has discussed several of the challenges in the design of high-speed CDR circuits and architectures, including jitter, skew and acquisition of lock.

Lock acquisition for a CDR design used in spread spectrum SerDes applications such as the Serial AT Attachment (SATA) [2] is particularly challenging because the data rate varies periodically. Moreover, jitter and skew are generated during the data transmission. An example of time domain spread spectrum clocking (SSC) based on the serial-ATA Gen. II specification is shown in Figure 1 [2]. The purpose of this type of clocking is to spread the spectral energy and thereby reduce the effect of electromagnetic interference in the electronic system. The frequency moves between 0 ppm and –5000 ppm, which is known as a down-spread spectrum [2]. Since the period of the spread spectrum modulation is between 30 and 33 kHz, the time for acquisition of lock must be much less than 30-33 uS. This is required in order to minimize the jitter or phase noise due to the periodic variation of the data rate. A possible transceiver block diagram with spread spectrum clocking is shown in Figure 2. The SSC clock is used in the transmitting path and extracted in the receiving path.

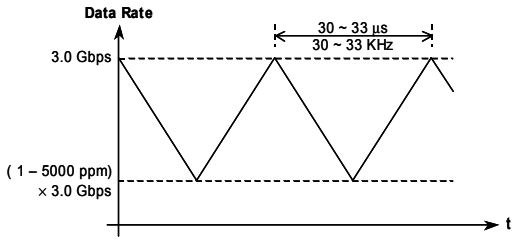

Figure 1. Time domain spread-spectrum clocking operation.

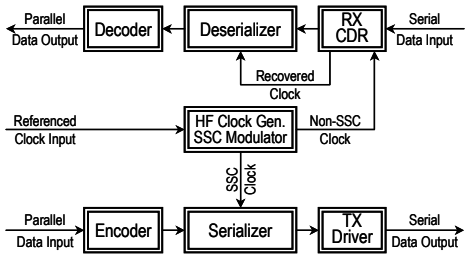

Figure 2. Block diagram of a spread-spectrum SerDes.

II. CDR ARCHITECTURE

The proposed architecture uses a phase interpolator based CDR design including a half-rate Alexander type phase detector [3] and a phase-shifting type phase interpolator [4] with a frequency differentiator and an adaptive loop gain filter. Figure 3 shows the basic block diagram of the proposed CDR architecture. First, the frequency differentiator periodically determines the clock rate difference between the referenced clock and the recovered clock. This difference in clock rates is then used to periodically adjust the gain of an adaptive loop filter for better acquisition of lock. In the case of a large difference in the two clock rates, a higher loop gain is needed for quick acquisition of lock. However, for a smaller difference of

clock rates, a lower loop gain is needed for minimized jitter and phase noise.

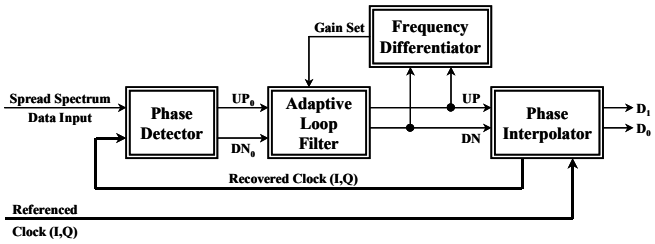

Figure 3. A block diagram of the proposed CDR architecture.

A. Phase Detector

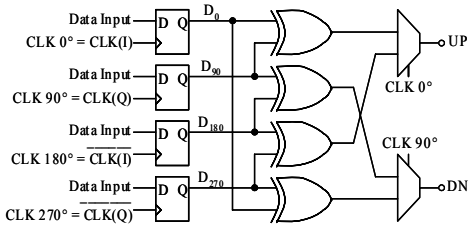

Figure 4. Half-rate Alexander type phase detector.

The half-rate Alexander type phase detector shown in Figure 4 [3] samples the data at 0°, 90°, 180° and 270° of the clock signal and produces the corresponding D_0, D_{90}, D_{180} and D_{270} signals. The outputs UP and DN are then generated from the XOR gates and 2-to-1 multiplexers. Once the CDR is in the locked state, the quadrature clock edges are aligned with the received data transitions. Therefore, D_0 and D_{180} are the recovered, demultiplexed data streams.

B. Phase Interpolator

The phase-shifting type phase interpolator shown in Figure 5 [4] is used, where I_1, I_2, I_3 and I_4 are controlled by a shift register. At any given time, only one-fourth of the consecutive shift register stages have logic "high" which turns on the tail current of the phase interpolator. For a data rate variation greater than 0 ppm, the shift register stages with logic "high" will shift to the left to ramp up the recovered clock frequency and vice-versa. The recovered clock voltage signal from the phase interpolator output is shown in equation (1)

$$Vout_{PI} = RL \cdot (I_1 + I_2 + I_3 + I_4) \tag{1}$$

where RL is the load resistance. For every received UP pulse, it shifts the data to the left and increases the average recovered clock frequency. For every received DN pulse, it shifts the data to the right and decreases the average recovered clock frequency.

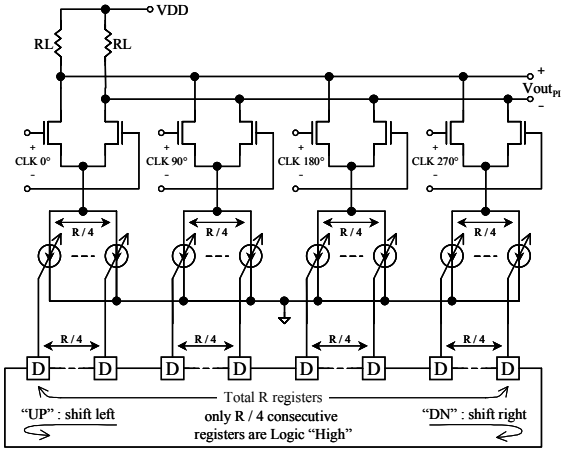

Figure 5. Implementation example of phase interpolator.

C. Frequency Differentiator

The frequency differentiator design is based on a digital integrator which adds one for every UP pulse and subtracts one for every DN pulse. Assume that there are total of *R* registers in the phase interpolator. Shifting registers with logic "high" to the left *R* times in every 1000 reference clocks causes the recovered clock phase to shift down by one referenced clock period. In other words, shifting the recovered clock phase down for one referenced clock period every 1000 referenced clocks moves the average recovered clock frequency up by 1000 ppm and vice-versa. Hence, the clock rate difference between the referenced clock and the recovered clock for a given time period can be calculated as follows:

$$\Delta f(CLK_{CDR}) = \frac{f(CLK_{CDR}) - f(CLK_{REF})}{f(CLK_{REF})} = \frac{P/R}{C} \tag{2}$$

In this equation, *C* is the number of referenced clock periods for a given time, *P* is the total number of UP pulses minus the total number of DN pulses, *R* is the number of shift registers stages used in the phase interpolator, $f(CLK_{CDR})$ is the frequency of the recovered clock, $f(CLK_{REF})$ is the frequency of the referenced clock and $\Delta f(CLK_{CDR})$ is the frequency variation of the recovered clock.

In order to have the output of the frequency differentiator track the spread spectrum frequency variation, the system must reset the frequency differentiator periodically in a time much less than spread spectrum modulation period.

D. Adaptive Loop Gain Filter

The gain of the adaptive loop filter is set based on the output of the frequency differentiator. The loop filter removes some of the UP and DN pulses depending on the value of the gain setting. If the recovered clock frequency is moving up away from the reference clock frequency, the adaptive loop filter will let more UP pulses into the phase interpolator. On the other hand, if the recovered clock frequency is moving down away from referenced clock

frequency, it will allow more DN pulses into the phase interpolator.

This proposed CDR architecture is designed to handle ±5000 ppm of spread spectrum data variation with a minimum of 0.5 UI deterministic jitter tolerance and a maximum of 0.1 UI internally generated jitter. In order to meet these requirements, the circuit has 80 shift register stages in the phase interpolator and seven levels of gain in the adaptive loop filter. In the half-rate Alexander type phase detector, one bit of the data period is equal to one-half of the recovered clock period. In other words, the phase step is 2 UI ÷ 80 = 0.025 UI, which can handle a maximum 4-step phase shifting due to internal jitter in the locked state and still meet the maximum 0.1 UI jitter generation requirement.

TABLE I. GAIN SETTING OF THE FIXED LOOP FILTER

Fixed Loop Filter Gain	Δf(Data) Range ppm	"UP"			"DN"		
		Filter Pass/Block Ratio	Maximum Δf(Clcok) ppm	Maximum Phase Error ppm	Filter Pass / Block Ratio	Maximum Δf(Clcok) ppm	Maximum Phase Error ppm
0	+ 5000 ~ - 5000	1 / 1	+ 6250	11250	1 / 1	- 6250	11250

TABLE II. GAIN SETTING OF THE ADAPTIVE LOOP FILTER

Adaptive Loop Filter Gain	Δf(Data) Range ppm	"UP"			"DN"		
		Filter Pass/Block Ratio	Maximum Δf(Clcok) ppm	Maximum Phase Error ppm	Filter Pass / Block Ratio	Maximum Δf(Clcok) ppm	Maximum Phase Error ppm
+ 3	+ 5000 ~ + 4000	1 / 1	+ 6250	1250	1 / 14	- 833	5833
+ 2	+ 4000 ~ + 2400	2 / 3	+ 5000	1000	1 / 14	- 833	4833
+ 1	+ 2400 ~ + 800	1 / 3	+ 4167	1767	1 / 14	- 833	3233
0	+ 800 ~ - 800	1 / 4	+ 2500	1700	1 / 4	- 2500	1700
- 1	- 800 ~ - 2400	1 / 14	+ 833	3233	1 / 3	- 4167	1767
- 2	- 2400 ~ - 4000	1 / 14	+ 833	4833	2 / 3	- 5000	1000
- 3	- 4000 ~ - 5000	1 / 14	+ 833	5833	1 / 1	- 6250	1250

Tables I and II show the input data rate variation range, the maximum recovered clock rate tracking and the phase error between data and clock for fixed and adaptive loop filters, respectively The gain setting in the adaptive loop filter is based on the minimum required gain for the recovered clock frequency to track the spread spectrum data rate and still meet the maximum allowed internal jitter generation and minimum deterministic jitter tolerance. The maximum Δf(clock) and phase error for a given gain setting are calculated in the following equations:

$$\Delta f\,(Clock)[ppm] = \frac{S_P}{S_P + S_B} \cdot \frac{10^6}{R} \tag{3}$$

$$PE[ppm] = |\Delta f\,(Clock) - \Delta f\,(Data)| \tag{4}$$

$$PE[UI] = PE[ppm] \cdot Delay[UI] \tag{5}$$

where S_P and S_B are the pass and block values from Tables I and II, R is the number of shift register stages in the phase interpolator and PE[ppm] and PE[UI] are the phase error in ppm and Unit Intervals (UI), respectively. One UI is equal to one bit of the data period and Delay[UI] is the CDR loop delay in UI. For this design, the loop delay is 8 UI.

III. CALCULATION AND SIMULATION RESULTS

Figures 6 and 7 show the calculated phase shift and phase error for fixed loop gain and adaptive loop gain CDR designs, respectively. The calculation is based on the maximum data rate variation and the available maximum recovered clock frequency tracking values of Table I. Comparing the calculated phase errors between the fixed gain and adaptive gain, the fixed gain setup has twice as much phase error as the adaptive gain case. This is due to the extra UP and DOWN gain under positive and negative input data rate variations, respectively.

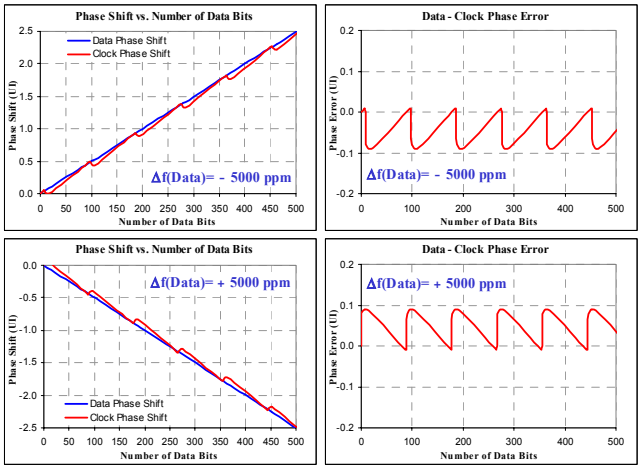

Figure 6. Caluclated phase shift and phase error for a fixed loop gain CDR design.

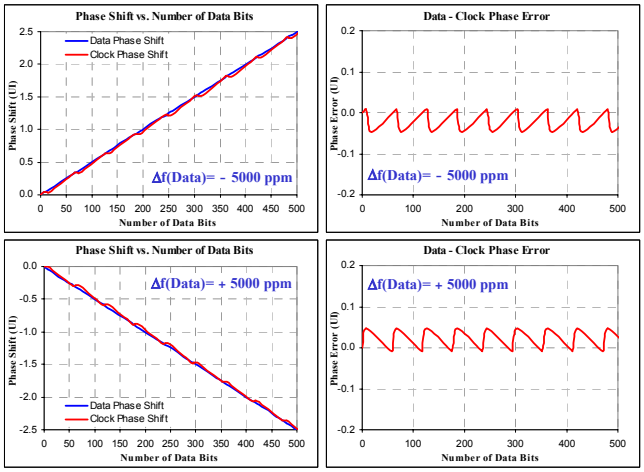

Figure 7. Calculated phase shift and phase error for an adaptive loop gain CDR design.

Figures 8 and 9 show the simulation results for the phase shift and phase error in the fixed loop gain and adaptive loop gain CDR designs, respectively. The fixed gain scheme has twice as much phase error as the adaptive gain case, which was also predicted in the calculated results. The input data frequency versus time for phase shift and phase error simulations are also shown in these figures. Figure 9 for the adaptive loop gain CDR design shows how the internal gain tracks the input data rate variation.

Figure 10 shows eye diagrams of the input data, the retimed data, the referenced clock and the recovered clock at –3000 ppm of the normal data rate with 0.5 UI jitter associated with the data for the adaptive loop gain CDR design.

The input data shown Figure 10 has deterministic jitter jumping around at the ±0.25 UI and 0 UI points for the 3.0 Gbps data rate. The retimed data is one of two demultiplexed data streams from the phase detector and has a data rate 1.5 Gbps. The simulation results show the removal of deterministic jitter associated with the input data. The only residue jitter in the retimed data is the CDR internally generated jitter from the tracking phase error between the input data and the recovered clock.

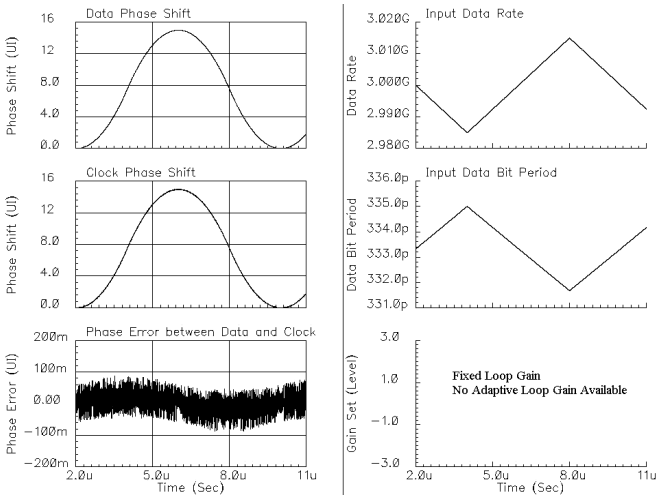

Figure 8. Simulation results of phase shift and phase error for fixed loop gain CDR design.

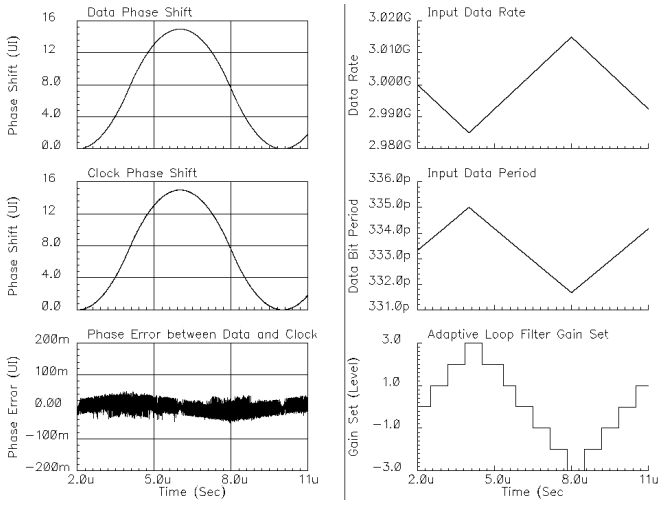

Figure 9. Simulation results of phase shift and phase error for adaptive loop gain CDR design.

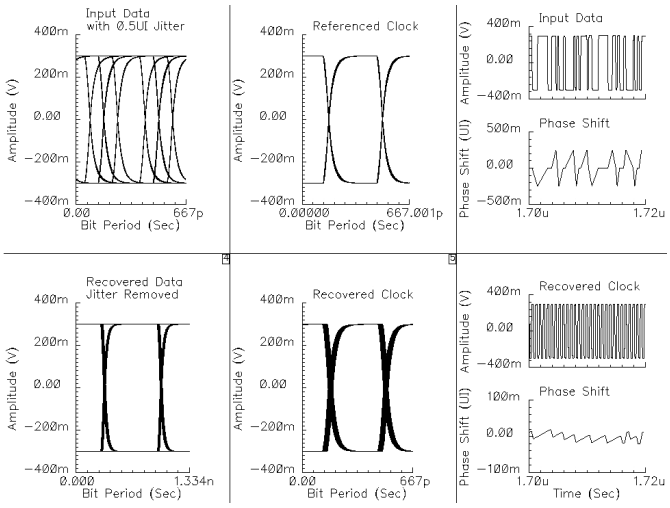

Figure 10. Eye diagrams of the input data, retimed data, referenced clock and recovered clock at –3000 ppm of the normal data rate with 0.5 UI jitter associated with the data for the adaptive loop gain CDR design.

IV. CONCLUSION

A requirement for spread-spectrum clocking exists in some SerDes applications such as the Serial AT Attachment. This reduces the effect of electromagnetic interference in the electronic system but presents challenges in the CDR design. The proposed phase interpolator based CDR with adaptive loop gain fulfills the need for better acquisition of lock with only a small amount of internally generated jitter. Both calculated and simulated results show that the use of adaptive gain in the CDR design eliminates half of the internally generated jitter from the tracking phase error between the input data and the recovered clock and still meets the requirement for high deterministic jitter tolerance.

REFERENCES

[1] B. Razavi, “Challenges in the Design of High-Speed Clock Data Recovery Circuit,” IEEE Communication Magazine, August 2002, pp 94-101.

[2] Serial ATA Working Group, “Serial ATA II: Electrical Specification, Revision 1.0”, Serial ATA Working Group http://www. sata-io.org, May 2004.

[3] A. Rezayee, and K. Martin, "A 9-16Gb/s Clock and Data Recovery Circuit with Threestate Phase Detector and Dual-Path Loop Architecture," Proceedings of the 2003 European Solid-State Circuits Conference, Estoril, Portugal, pp. 683-686.

[4] R. Kreienkamp, U. Langmann, C. Zimmermann, T. Aoyama, “A 10-Gb/s CMOS clock and data recovery circuit with an analog phase interpolator,” Proceedings of the IEEE 2003 Custom Integrated Circuits Conference, pp. 73-76.