A Fast-locking Clock and Data Recovery Circuit with A Lock Detector Loop

Chih-Lin Chen, *Student Member, IEEE*,
and Chua-Chin Wang†, *Senior Member, IEEE*

Chun-Ying Juan

Department of Electrical Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan 80424
Email: ccwang@ee.nsysu.edu.tw

Metal Industries Research & Development Centre (MIRDC),
Taipei 106, Taiwan.
Email: chunying@mail.mirdc.org.tw

Abstract—**This work presents a PLL-based (phase-locked loop) clock and data recovery (CDR) circuit with a lock detector loop for fast locking and low jitter. We use an adjustable charge pump to change the charge current according to the state of the lock detector loop, which is determined by seven clocks with equal phase difference. An experimental prototype was implemented using a typical 0.18 μm CMOS process. The post-layout-extracted simulation results reveal that the worst case jitter of the recovery clock is less than 199.66 ps (peak-to-peak) and the settling time is less than 4 μs at all PVT (Process, voltage, and temperature) corners.**

Index Terms—**fast-locking, phase shift, CDR, and lock detector loop.**

I. INTRODUCTION

Recently, FlexRay [1] is considered as a total solution to be integrated with different in-car communication specifications, i.e., CAN and MOST. FlexRay is mainly aimed at safety and reliability. In next-generation FlexRay specification, the data rate might be drastically increased for adding more audio/video equipments in a vehicle, e.g., mobile TV receiver, GPS, video player, video game, and so on. Therefore, clock and data recovery (CDR) circuit will be required in the future FlexRay systems.

Two major architectures for CDR designs are PLL-based CDR and phase interpolation CDR. The former is close to a PLL architecture, including phase/frequency detector (PFD), voltage controlled oscillator (VCO), charge pump (CP), and low pass filter (LPF). However, the disadvantages include: the clock frequency of VCO can not surpass ±50% of the center frequency [2], the settling time is long depending on bandwidth and jitter of the system, and the CDR circuit might be locked in a wrong frequency called harmonic lock. Traditionally, FlexRay systems need a high frequency (8 times data rate) phase-locked loop (PLL) circuit to achieve over-sampling function and data recovery. For example, if FlexRay system operates at a high date rate (e.g., 100 Mbps), a 800 Mbps PLL is required for over-sampling. A reliable 800 Mbps PLL is not easy to be designed and integrated in system-on-

chip (SOC) such that it is not reliable to be used in a in-vehicle network for the sake of safety.

Phase interpolation CDR utilizes the output clock of VCO with different phases to sample data, and then the sampled results are transferred to recover data. Notably, each bit of the data only needs 3 sampled bits to recover. Meanwhile, the jitter performance depends on equalization of the clock phase shift generated by VCO. Notably, it is also very hard to have equal phase shift between two adjacent clocks.

CDR is mainly used to generate clock, synchronize received data, and reduce jitter. In a receiver design, receivers need a CDR circuit to synchronize data with the clock, because incoming data are usually asynchronous with respect to the system clock. If the incoming data are coupled with noise, the receiver with CDR should reject noise to reduce jitter. In prior reports, CDR designs usually had a trade-off between settling time and clock jitter for different applications. If a CDR circuit operates in a short settling time, it will have a poor jitter performance. By contrast, a low-jitter CDR circuit must spend longer time to lock the incoming data. To achieve short settling time and low jitter, a lock detector was reported to adjust loop bandwidth in a receiver system [3]- [7]. PLL designs [3] use a digital frequency difference detector (DFDD) [4] to adjust resistors of LPF to change system bandwidth for short settling time. In prior CDR designs [5]- [6], a lock detector was proposed to change system bandwidth for fast locking. Another kind of CDR design [7] utilized a lock detector to detect the transition with respect to a reference clock. If the clock transition occurs before or after the reference clock, a counter is counted up. Otherwise, the counter is counted down. The counter then determines the CDR circuit to operate in a frequency detecting loop or a phase detecting loop.

This work proposes a 100 Mbps CDR circuit with short locking time and low jitter. The proposed CDR circuit is used to recover data bits given a 100 Mbps data rate for FlexRay specifications. In Section II, we introduce the architecture of PLL-based CDR with a lock detector loop and show the data flow diagram. In Section III, we demonstrate the simulation results of CDR circuit by MATLAB and HSPICE. We compare our CDR circuit with the prior works in a comparison table

---

†: Prof. C.-C. Wang is the contact author.

as well. A brief conclusion is given in Section IV.

II. ARCHITECTURE

The proposed CDR circuit includes a phase detector (PD), a frequency detector (FD), two charge pumps for PD and FD (CP_PD and CP_FD), a second-order low-pass filter (R1, C1, and C2), a voltage controlled oscillator (VCO), a Divider, and a lock detector loop, as shown in Fig. 1. The function of each block is given in the following text.
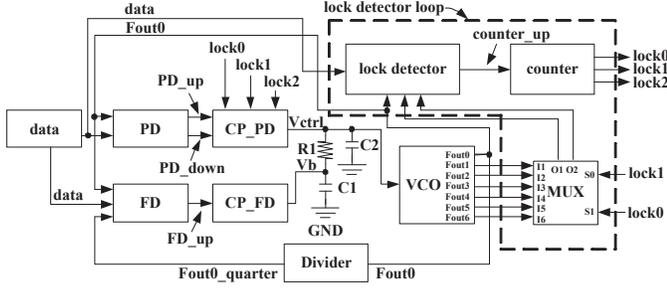

Fig. 1. The proposed CDR circuit with a lock detector loop

A. Phase detector (PD)

The Alexander binary phase detector in [8] is used as our PD. If the data transition occurs after the Fout0's falling edge, the PD_up outputs logic '1' to increase clock frequency of VCO, where Fout0 is the main base output clock. Otherwise, if the data transition occurs before the Fout0's falling edge, the PD_up outputs logic '0' to decrease clock frequency of VCO. Because the PD does not have a linear frequency response, the PD causes a constant jitter in the CDR system.

B. Frequency detector (FD)

The FD in [9] is used to detect whether the data stream have two rising edges during Fout0_quarter is logic '0' or logic '1', where the frequency of Fout0_quarter is a quarter of the output clock frequency of VCO that is generated by the Divider. If the FD detects two rising edges, FD_up outputs logic '1'. In other words, the clock frequency of VCO is less than 2 times of data rate such that the clock frequency of VCO must be increased.

C. Charge pump for PD and FD (CP_PD and CP_FD)

Fig. 2 shows the schematic of the charge pump for PD (CP_PD). R_{PD} is used to generate a bias current, I_{RPD}. Notably, I_P is determined by I_{RPD}, lock0, lock1, and lock2. The maximum charge current, I_P, is 8 times of I_{RPD}. If PD_up is logic '1', I_P flows into Vctrl from VDD. If PD_down is logic '1', I_P flows into GND from Vctrl. The architecture of the charge pump for FD (CP_FD) is identical to that for PD.
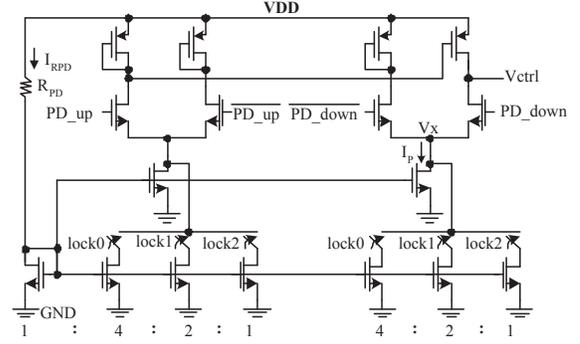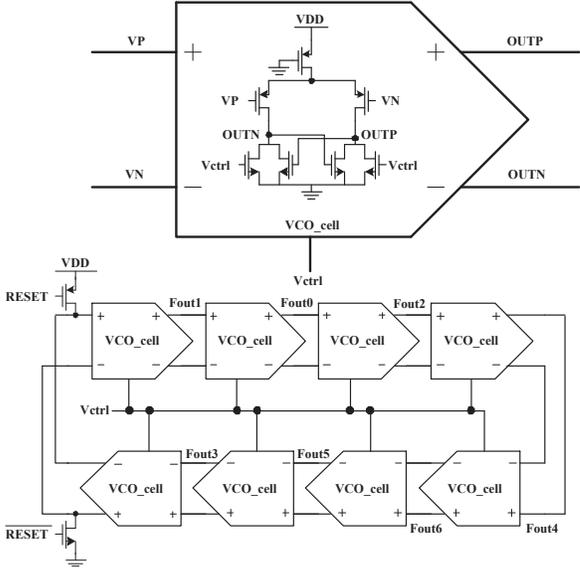

Fig. 2. The schematic of CP_PD

Fig. 3. The architecture of VCO and the schematic of VCO cell

D. Voltage controlled oscillator (VCO)

Fig. 3 shows the VCO block diagram, where the schematic of the VCO_cell is included. A differential 8-stage voltage controlled ring oscillator is used to generate clocks with equal phase shift. When RESET is activated, the VCO starts oscillating. Vctrl is used to adjust the clock frequency of VCO. VCO_cells accumulate different phase delays, respectively, to generate a bank of clocks with seven different phases, i.e., Fout0 to Fout6. Fout0 is the main clock of VCO to be synchronized with the data. Fout1, Fout3, and Fout5 lead Fout0 in phase, while Fout2, Fout4, and Fout6 lag in phase.

E. Lock detector loop

The lock detector loop is composed of a lock detector, a MUX, and a counter. Three D-flip-flops (DFFs) and an XNOR are used to implement the lock detector, as shown in Fig. 4. The lock detector is used to detect if the positive and negative edges of incoming data are synchronized with Fout0. Referring to Fig. 1, lock0 and lock1 are used to select different clocks fed into the lock detector by the MUX, which are described as follows.
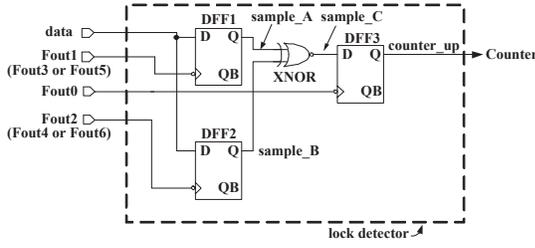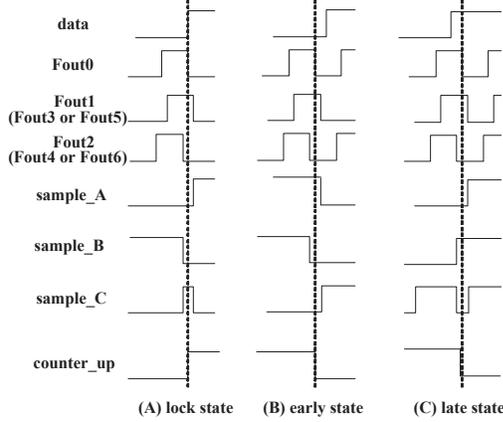
Fig. 4. The architecture of lock detector

Fig. 5. Three different scenarios in the lock detector loop

Fig. 6. The clocks with equal phase shift

- If {lock0, lock1}='00', MUX selects Fout1 and Fout2 fed into the lock detector.
- If {lock0, lock1}='10', MUX selects Fout3 and Fout4 fed into the lock detector.
- If {lock0, lock1}='11', MUX selects Fout5 and Fout6 fed into the lock detector.

Fig. 5 shows the three scenarios in the lock detector loop, which are described as follows. DFF1 uses three clocks, Fout1, Fout3, and Fout5, to sample data depending on lock0 and lock1. If the data transition occurs before Fout1, sample_A is set to logic '1'. Otherwise, if the data transition occurs after Fout1, sample_A is set to logic '0'. DFF2 uses three clocks, Fout2, Fout4, and Fout6, to sample data also depending on lock0 and lock1. If the data transition occurs after Fout2, sample_B is set to logic '1'. Otherwise, if the data transition occurs before Fout2, sample_B is set to logic '0'. If CDR circuit is the lock state, the data transition occurs after Fout2 and before Fout1. In other words, sample_A and sample_B are switched to logic '0', as shown in Fig. 5(A). By contrast, if CDR circuit is not locked, sample_A and sample_B are either switched to logic '0', as shown in Fig. 5(B) or Fig. 5(C). Sample_A and sample_B are coupled to the inputs of XNOR to decide which scenario occurs. DFF3 is used to sample the output of XNOR, sample_C, by Fout0. The sampled result is "counter_up", which is used to trigger the counter or reset it.

The phase shift of each clock is shown in Fig. 6. Because the CP_PD initially operates with a large current for short settling time, the clock jitter is large. In other words, the phase shift of any two adjacent clocks is large. To enhance the reliability
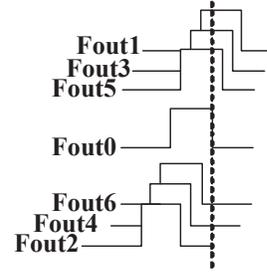
of the lock detector, the MUX selects Fout1 and Fout2 for the lock detector in the beginning, because the phase difference between Fout1 and Fout2 is maximum. Therefore, the phase shift between Fout1 and Fout2 can cover variation of the clock jitter given a wrong detection. If the CDR circuit continues to be in the lock state, the counter continues to count up. If the counter runs up to a pre-defined value, lock0 is switched to logic '1' to ensure that CDR circuit is locked. Therefore, the current of CP_PD is decreased to reduce the clock jitter. Then, the MUX selects Fout3 and Fout4 for the lock detector. If the counter runs up to a pre-defined value, lock1 is switched to logic '1' to reduce more current of CP_PD. Finally, the MUX selects Fout5 and Fout6 for the lock detector. Similarly, if the counter runs up to a pre-defined value, lock2 is switched to logic '1' to further reduce more current in CP_PD. In short, depending on lock0, lock1, and lock2, the current of CP_PD is reduced step by step such that the clock jitter behaves the same.

III. SIMULATION RESULTS

For the sake of stability, we need to avoid CDR system from oscillating such that the phase margin of the the CDR system must be larger than 0°. Eqn. (1) is the transfer function of our CDR shown in Fig. 1, which is simulated using MATLAB to derive all of the system parameters given in Table II. By Eqn. (1), the phase margin is 71° and the bandwidth is 13.7 MHz.

$$\text{Frequency response} =$$
$$I_{RPD} \times \text{VCO gain} \times \frac{1}{s} \times \frac{R1 + \frac{1}{s \times C1}}{s \times C2 \times (R1 + \frac{1}{s \times C1} + \frac{1}{s \times C2}))} \quad (1)$$

The proposed design is implemented using a typical 0.18 μm CMOS process to justify the performance. Notably, all of the process corners: [-40^oC, $+125^o$C] and [SS, TT, FF] models are simulated. Fig. 7 shows the transient response between Vctrl and the lock detector. The voltage amplitude of Vctrl is

|	Year	Process (μm)	Frequency (Mbps)	Clock jitter (pk-pk)	Settling time (bits/time)	Power (mW)	Supply voltage	Core area (μmm^2)	FOM $\frac{\text{Mbps}\times\mu\text{m}\times\text{V}^2}{\text{bits}\times\text{mW}\times\text{ps}\times\text{mm}^2}$
Ours	2011	0.18	100	199.66 ps	400 bits/4μs	6.649	1.8 V	0.16	686.42
[7]	2009	0.18	5120/6400	2.12 ps(rms)	N/A	136	1.8 V	0.8	N/A
[11]	2008	0.18	662-3125	62.2 ps	>331000 bits / >500 μs	60	1.8 V	0.1326	118.8
[5]	2006	0.18	2500	88 ps	4862 bits/3.89 μs	N/A	1.8 V	0.133	N/A
[10]	2006	0.18	155.52-3125	467 ps	>15552 bits/ >100μs	95	1.8 V	0.88	3
[12]	2006	0.35	200-2000	120 ps	>60000 bits/ >30 μs	170	3.3 V	0.4	15.56
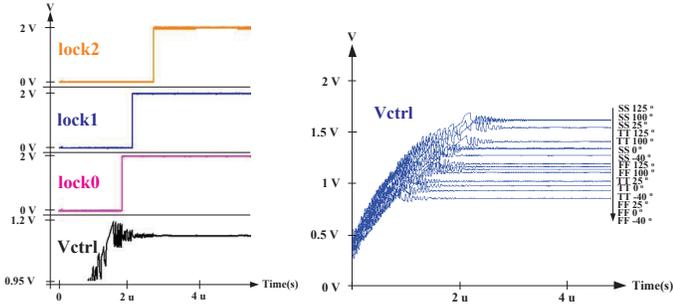
Fig. 7.   Simulation waveform of the lock detector
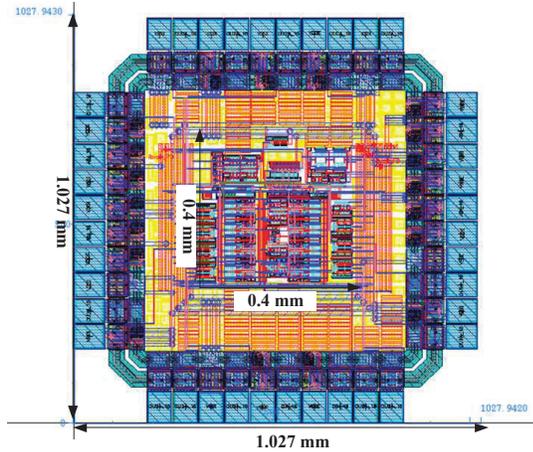

Fig. 8.   Simulation waveform of Vctrl

Fig. 9.   Layout of the proposed design

decreased from 80 mV to 0.3 mV. Fig. 8 shows the simulation waveforms of Vctrl at different process corners. Fig. 9 shows the layout of the proposed design. The chip area is 1.027×1.027 mm^2, and the core area is 0.4×0.4 mm^2. Table I shows the comparison between the proposed design and several prior works. Besides attaining the least power dissipation and the shortest settling time, our design also shows the best FOM.

IV. CONCLUSION

The proposed CDR circuit resolves the difficulty of traditional CDR designs, because it does not need to make a trade-off between settling time and jitter. When CDR circuits include a lock detector loop, they can use a larger current in charge pumps to shorten settling time. When CDR circuits are locked, they decrease the current in the charge pumps to reduce jitter.

Therefore, our proposed design has a good performance in both settling time and jitter.

REFERENCES

[1] FlexRay Communications System - Protocol Specification V2.1 (http://www.flexray.com), 2005.

[2] R. J. Baker, H. W. Li, and D. E. Boyce, CMOS Circuit Design, Layout, and Simulation, 2nd ed. New Jersey: John Wiley & Sons, Inc., 1997.

[3] Y. Tang, M. Ismail, and S. Bibyk, "A new fast-settling gearshift adaptive PLL to extend loop bandwidth enhancement in frequency synthesizers," in Proc. IEEE International Symposium on Circuits and Systems, May 2002, vol. 4, pp. 787-790.

[4] I. Hwang, S. Lee, and S. Kim, "A digitally controlled phase loop with fast locking scheme for clock synthesis application," in Proc. IEEE International Solid-State Circuits Conference Digest Technical Papers, Feb. 2000, pp. 168-169.

[5] J.-K Woo, H. Lee, W.-Y. Shin, H. Song, D.-K. Jeong, and S. Kim, "A fast-locking CDR circuit with an autonomously reconfigurable charge pump and loop filter," in Proc. IEEE Asian Solid-State Circuits Conference, Nov. 2006, pp. 411-414.

[6] J.-K Woo, D.-K. Jeong, and S. Kim, "Fast-locking CDR circuit with autonomously reconfigurable mechanism," Electronics Letters, vol. 43, no. 11, pp. 624-626, May 2007.

[7] F.-T. Chen and J.-M. Wu, "An extended phase detector 2.56/3.2Gb/s clock and data recovery design with digitally assisted lock detector," in Proc. IEEE International Symposium on Circuits and Systems, May 2009, pp. 1831-1834.

[8] J. D. H. Alexander, "Clock recovery from random binary signals," Electronics Letters, vol. 11, no. 22, pp. 541-542, Oct. 1975.

[9] D. Dalton, K. Chai, E. Evans, M. Ferriss, D. Hitchcox, P. Murray, S. Selvanayagam, P. Shepherd, and L. DeVito, "A 12.5-Mb/s to 2.7-G/s continuous-rate CDR with automatic frequency acquisition and data-rate readback," IEEE Journal Solid-State Circuits, vol. 40, no. 12, pp. 2713-2725, Dec. 2005

[10] R.-J. Yang, K.-H. Chao, S.-C. Hwu, C.-K. Liang, and S.-I. Liu, "A 155.52 Mbps-3.125 Gbps continuous-rate clock and data recovery circuit," IEEE Journal Solid-State Circuits, vol. 41, no. 6, pp. 1380-1390, Jun. 2006.

[11] S.-H. Lin and S.-I. Liu, "Full-rate bang bang phase/frequency detectors for unilateral continuous-rate CDRs," IEEE Transactions on Circuits and Systems II, vol. 55, no. 12, pp. 1214-1218, Dec. 2008.

[12] R.-J. Yang, K.-H. Chao, and S.-I. Liu, "A 200-Mbps 2-Gbps continuous-rate clock-and-recovery circuit," IEEE Transactions on Circuits and Systems I, vol. 53, No. 4, pp. 842-847, Apr. 2006.