Design of 2.5-Gb/s Parallel PRBS Generator and 4-Gb/s Area Efficient PRBS Checker in 65-nm CMOS Process

Jun-Cheol Lee[1] and Joo-Hyung Chae[2, a]
[1]Department of Electrical Engineering, Kwangwoon University
[2]Department of Electronics and Communications Engineering, Kwangwoon University
E-mail: [1]ldojj@kw.ac.kr

***Abstract* - A pseudo-random binary sequence (PRBS) pattern is widely used in various applications requiring random data sequences. This paper describes the design of a parallel PRBS generator and checker. The parallel PRBS generator produces eight outputs where each output has a PRBS-7 pattern with a period of 2^7–1, and the PRBS checker verifies if the incoming data is a PRBS-7 pattern. In the PRBS checker, a method of creating the same reference PRBS pattern as the incoming data is applied using a synchronization detection circuit that finds a unique pattern in the input data stream. The PRBS checker includes data and error counters, and they count the number of input data and bit errors that occur during one cycle of the PRBS-7 pattern. This design was implemented with a 65-nm CMOS process, and the PRBS generator and checker occupy an area of 75×15 μm² and 75×45 μm², respectively. It was verified that the PRBS generator operated up to 2.5 Gb/s and the PRBS checker accurately counted the bit error at 4 Gb/s.**

***Keywords*—Bit error rate, pseudo-random binary sequence (PRBS), PRBS checker, PRBS generator**

I. INTRODUCTION

With various technology developments, the importance of a massive data movement continues to increase in high-speed serial interfaces, such as Universal Serial Bus (USB), Peripheral Component Interconnect Express (PCIE), Serial AT Attachment (SATA), and Ethernet. Furthermore, the recent trends of big data and artificial intelligence are explosively accelerating the demand for high I/O bandwidth [1]. However, channel insertion loss, noise, and crosstalk make increasing I/O bandwidth to be hard and increase a bit error rate (BER) [2]. Therefore, the BER check based on a pseudo-random binary sequence (PRBS) pattern is required to verify the quality of data transmission and reception.

When parallel interfaces were widely used for various standards of high-speed interface, a serial PRBS generator, as shown in Fig. 1(a), was widely used. However, as I/O bandwidth increased, a serializer/deserializer (SerDes) structure was adopted, which allowed the generation of high-speed off-chip serial data using low-speed on-chip parallel data. Accordingly, a parallel PRBS pattern generator, as shown in Fig. 1(b), is required. The parallel PRBS generator has the characteristic that the phase of each output is shifted from the other, so if serialized, a high-speed PRBS pattern can be created. Recently, more efficient structures of the parallel PRBS generator have been studied in terms of area occupation and power consumption 0, [3].

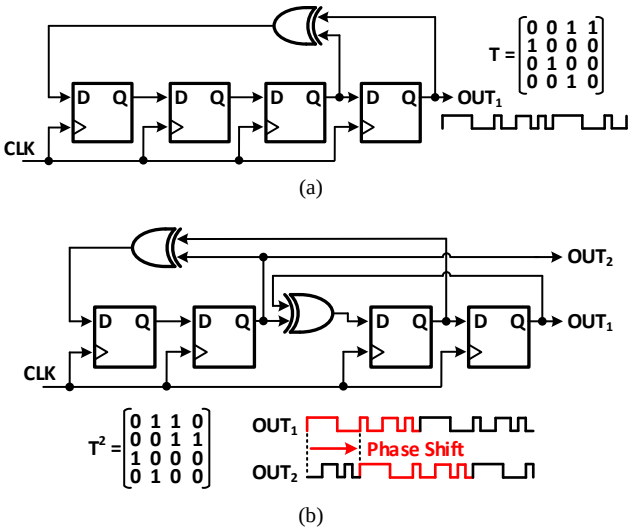

Fig. 1. (a) Serial PRBS generator and its transition matrix (T) and (b) 2-bit parallel PRBS generator and its transition matrix. Two generators show a PRBS-4 pattern generation case.

After the parallel PRBS pattern is serialized and transmitted, a receiver uses a deserializer to parallelize data received at high speed and then checks bit errors of the sampled data pattern using an on-chip PRBS pattern checker. Creating the same PRBS pattern based on the input PRBS pattern is called synchronization. Depending on the synchronization method, the PRBS checker can be implemented in two ways [4], [5], as shown in Fig. 2. The method shown in Fig. 2(a) occupies relatively small area, but if a bit error occurs, this error can give incorrect criteria to subsequent inputs, resulting in less accurate bit error test results: Fig. 2(a) also shows a case where a 1-bit error of the input PRBS data can result in inaccurate 3-bit error result.

a. Corresponding author; jhchae@kw.ac.kr

Manuscript Received Nov. 3, 2023, Revised Dec. 6, 2023, Accepted Dec. 6, 2023

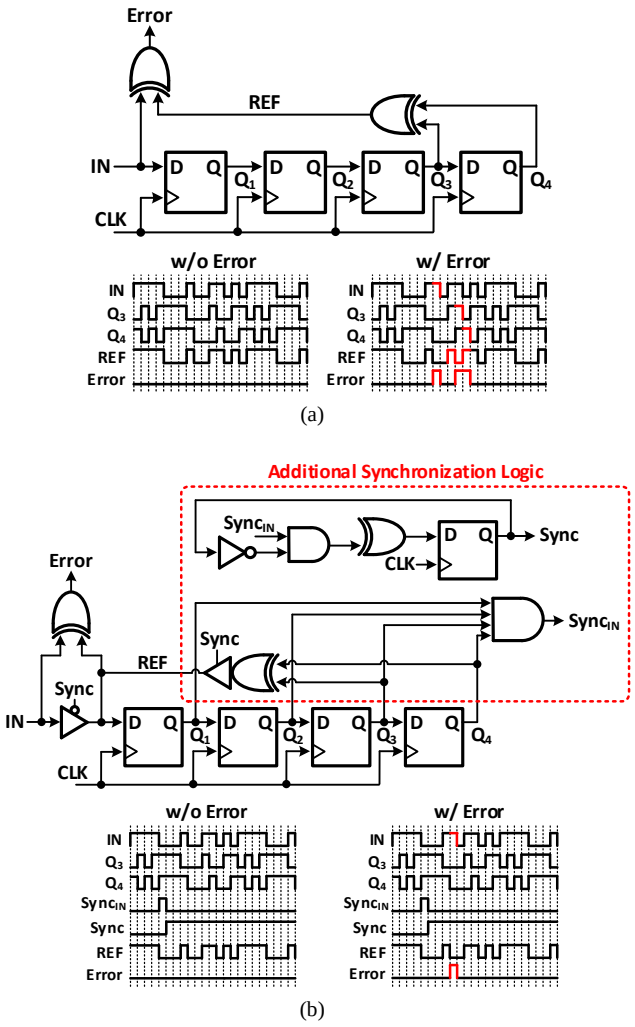

Fig.2. Two PRBS pattern checker implementations: (a) PRBS checker without synchronization and its operation with and without bit error; and (b) PRBS checker using an additional logic for precise error detection using synchronization.

The PRBS Checker in Fig. 2(b) has a larger area than that in Fig. 2(a) but is capable of precise error detection because it has an additional synchronization logic that can solve this problem.

We designed a PRBS-7 pattern generator with eight parallel outputs used as the input of SerDes, which includes an 8:1 serializer. In addition, a PRBS checker was designed utilizing a structure in Fig. 2(b). This can detect bit error accurately during one cycle of the pattern, enabling verification of the designed PRBS-7 generator. In order to count detected bit errors, a 7-bit counter was included, which allowed the BER check for one cycle of the PRBS-7 pattern. The designed PRBS checker can be extended to other PRBS patterns, such as PRBS-15, 23, and 31, as well as PRBS-7.

The remainder of the paper is organized as follows: Section II describes the architecture for the PRBS generator and checker; Section III shows post-layout simulation results for the PRBS generator and checker; and Section IV concludes the paper.

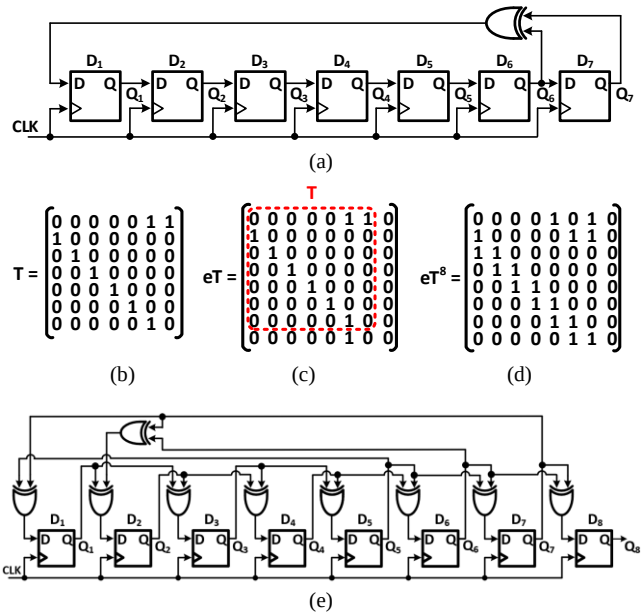

Fig. 3. (a) Block diagram of serial PRBS-7 pattern generator and (b) its transition matrix, (c) extension of transition matrix, (d) transition matrix for 8-bit parallel PRBS-7 generator, and (e) circuit diagram of 8-bit parallel PRBS-7 generator.

II. PRBS GENERATOR AND CHECKER DESIGN

A. Parallel PRBS-7 Generator with Eight Outputs

Fig. 3(a) shows a widely used PRBS-7 generator where the Q_6 and Q_7 signals are passed through the XOR logic and the output of the XOR is entered to the input of D_1. The PRBS pattern generation logic can be expressed through a transition matrix (T), and Fig. 3(b) shows the transition matrix of the PRBS generator shown in Fig. 3(a). Values "1" of the sixth and seventh columns in the first row express the XOR operation. A D_2 receives the output of D_1, which is expressed by the value "1" in the first column value of the second row of the transition matrix. The remaining D flip-flops (D-FF) were also expressed in the same way. If the value stored in D-FF in the current state is expressed as a vector U(j), the transition to the next state U(j+1) can be expressed as follows:

$$U(j+1) = T \cdot U(j). \quad (1)$$

Therefore, k transitioned state U(j+k) can be expressed as follows:

$$U(j+k) = T^k \cdot U(j) \quad (2)$$

Since the 8-bit parallel PRBS-7 generator has a larger number of the output bit than the serial PRBS-7 generator, an extended transition matrix (eT) should be made by adding "0"-filled single row and column, as shown in Fig. 3(c), before calculating the transition matrix for the parallel PRBS generator. Then, the eT^8 can be obtained from Equation (2), as shown in Fig. 3(d). Fig. 3(e) shows the circuit of the PRBS-7 generator with eight parallel outputs designed using eT^8.

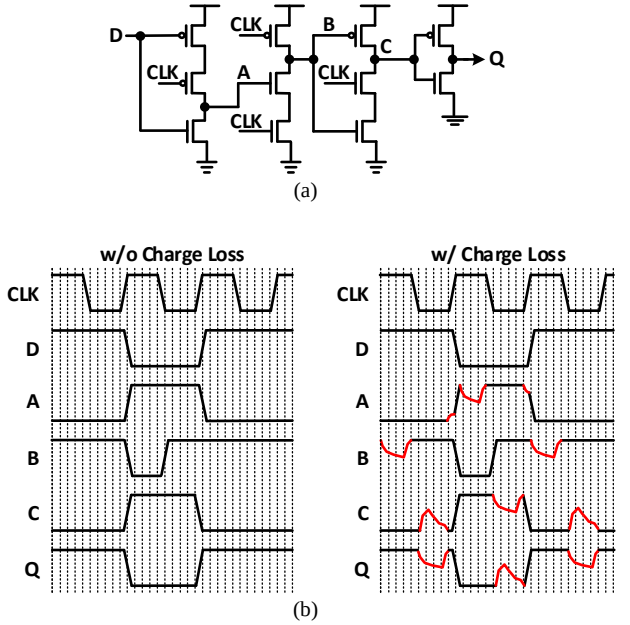

Fig. 4. (a) A TSPC D flip-flop and (b) its operation without and with charge loss.

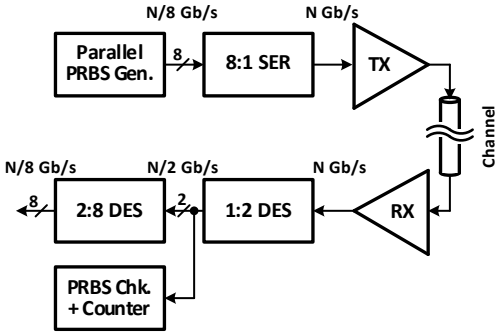

Fig. 5. Overall block diagram of SerDes to verify the PRBS generator (Gen.) and checker (Chk.).

B. Design Consideration for Selecting D-FF for PRBS Generator and Checker

The D-FF that constitutes the PRBS generator and checker can be designed in various methods, such as a true single-phase clock (TSPC) D-FF, a current-mode logic (CML) D-FF, and a transmission gate-based D-FF, depending on the design purpose [6], [7]. A CML D-FF can operate at the higher data rate than other D-FFs; however, it has the disadvantages of large area occupation and huge power consumption. A TSPC D-FF shown in Fig. 4(a) is also adopted for high-speed operations. It occupies relatively small area and consumes less power owing to its dynamic operation [8]. Fig. 4(b) shows an operation of the TSPC D-FF. In the high-frequency operation, data is stored in terms of electric charge in the parasitic capacitor of the metal line and the transistor. However, in the low-frequency operation, there can be a charge loss due to leakage current; thus, the operation can be failed. To assess the impact of leakage current in the 65-nm CMOS process, simulations were conducted in a worst-case environment, where the process, supply voltage, and temperature corners were set to SS, 0.9 V, and 90 °C, respectively. It was observed that the PRBS generator using TSPC D-FF did not operate properly when the clock frequency was below 130 MHz. However, a PRBS generator using the transmission gate-based D-FF under the same conditions operated correctly owing to its static operation.

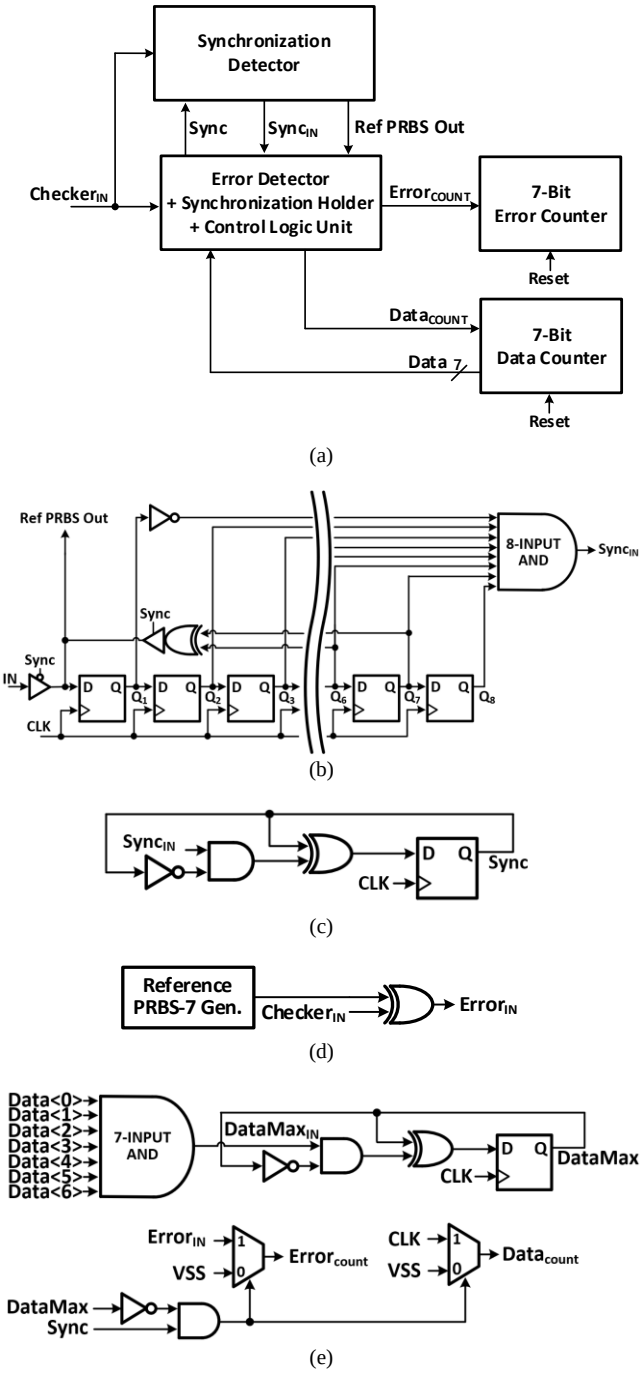

Fig. 6. (a) Block diagrams of PRBS checker, (b) synchronization detector, (c) synchronization holder, (d) error detector, and (e) stop logic of the counter.

Fig. 5 shows the overall block diagram of SerDes to verify the PRBS generator (Gen.) and checker (Chk.). The parallel PRBS generator makes the N/8-Gb/s 8-bit parallel data, and then the following 8:1 serializer (SER) makes N-Gb/s serialized data. Therefore, the parallel PRBS generator was

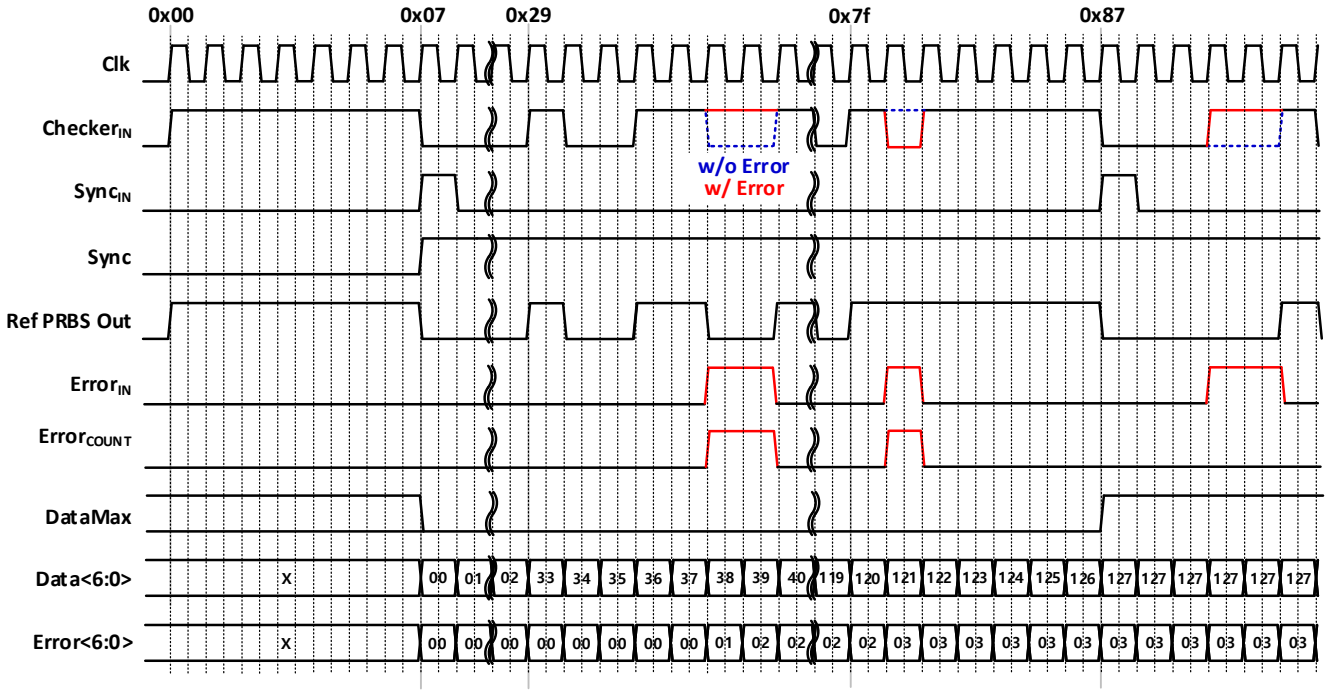

Fig. 7. PRBS checker operation.

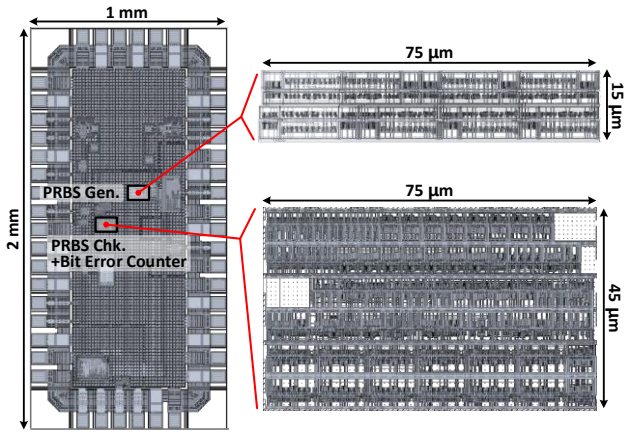

Fig. 8. Layout of SerDes including the PRBS generator and checker with the bit error counter.

designed using a transmission gate-based D-FF that can operate at a relatively low frequency. In the PRBS checker, a BER is checked using the output, operating at N/2 Gb/s, of the intermediate 1:2 deserializer (DES). The operation speed is higher than that of the PRBS generator. Therefore, the PRBS checker was designed using the TSPC D-FF, which occupies a smaller area than the transmission gate-based D-FF.

C. PRBS Checker with Bit Error Counter

The block diagram for the implemented PRBS checker is shown in Fig 6(a). As aforementioned, a separate synchronization circuit was used for precise error search. Fig. 6(b) shows the circuit implemented in accordance with the PRBS-7 pattern. In this circuit, $Sync_{IN}$ signal becomes HIGH when the pattern of the input signal, IN, is 11111110, which pattern occurs only once during a period of 2^7-1 bits of

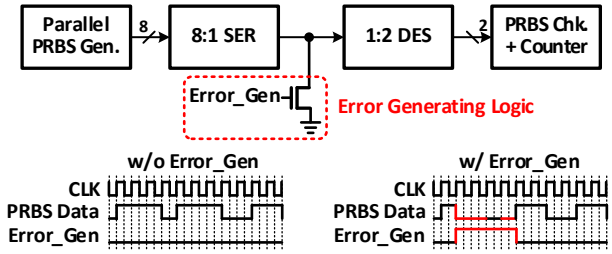

Fig. 9. Simulation setup for verifying the operation of the PRBS generator and checker.

PRBS-7. After $Sync_{IN}$ signal being HIGH, the serial PRBS-7 generator in the synchronization detector initializes Q7 through Q1 as 1111110, thereby synchronizing with the subsequent incoming PRBS-7 pattern. Therefore, the output of the synchronization detector can be used as a criterion for error judgment.

Fig. 6(c) shows a synchronization holder where the Sync signal with a continuous HIGH value is generated after the transition of the $Sync_{IN}$ signal is detected. When the Sync signal becomes HIGH, this signal is transmitted to the tristate buffer in the synchronization detector, shown in Fig. 6(b), and this detector operates as the reference PRBS-7 generator. Therefore, it does not require creating a separate PRBS-7 generator for checking the bit error, alleviating the area burden by reducing the number of flip-flops used. A circuit determining error can be implemented simply by using the XOR logic, as shown in Fig. 6(d). A 7-bit data and error counters were implemented to check BER during one cycle of the PRBS-7 pattern. After counting the one cycle of the PRBS-7 pattern, the counting operation should be stopped to prevent the overflow of the error count, and the stop logic of the counter is shown in Fig. 6(e).

Fig. 7 shows the operation of the PRBS checker; this also shows the error counting after the pattern synchronization

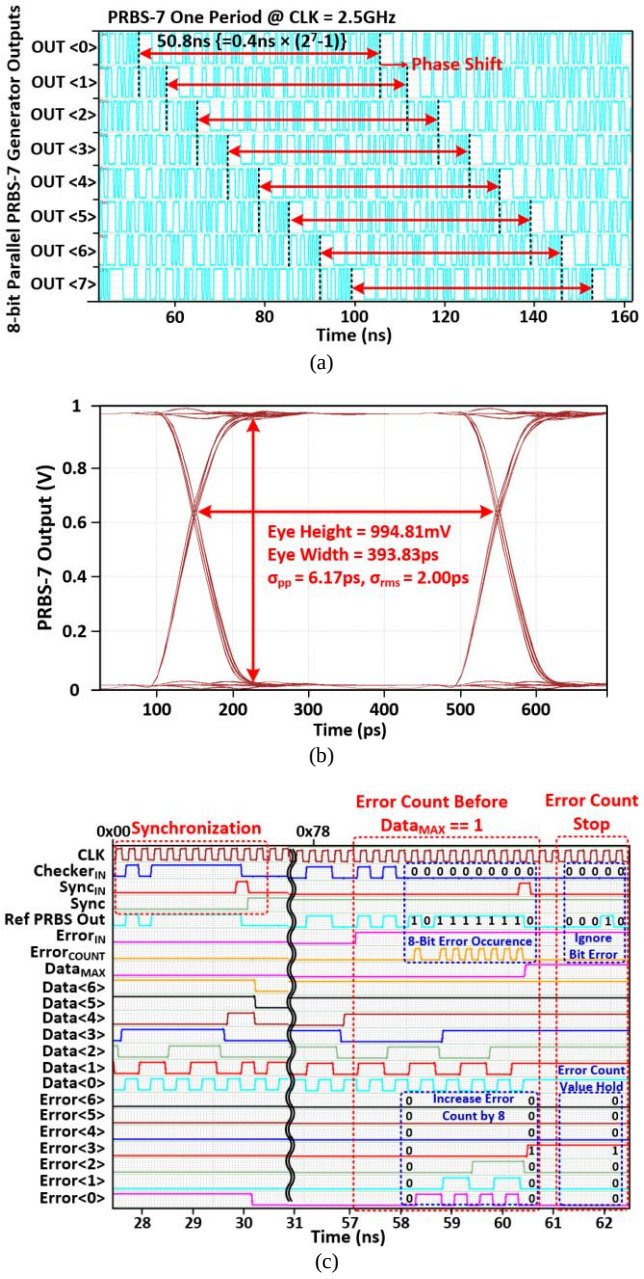

Fig. 10. Simulation results of (a) the 8-bit parallel PRBS-7 pattern outputs of the PRBS generator, (b) its output eye diagram, and (c) the PRBS checker and bit error counter.

and the hold operation of the final count value. When the data count is completed, the $Error_{COUNT}$ signal does not occur even if bit errors occur, maintaining the value of Error <6:0>.

III. SIMULATION RESULTS

The PRBS generator and checker, operating at a 1.0-V supply voltage, were implemented in a 65-nm CMOS process. Fig. 8 shows the layout of SerDes including the PRBS generator and checker with the bit error counter, in which the areas of the PRBS generator and checker are 75×15 μm^2 and 75×45 μm^2, respectively.

Fig. 9 shows a simulation setup for verifying the operation of the PRBS generator and checker. In order to generate bit errors, a single transistor is placed between the serializer and deserializer. When the HIGH Error_Gen signal is entered, a PRBS data having the HIGH value change to the LOW value, generating bit errors.

TABLE I. Performance Summary of PRBS Generator and Checker

	PRBS Generator	PRBS Checker
Technology [nm]	65	65
Supply [V]	1.0	1.0
Area [mm^2]	0.001125	0.003375
Peak-to-Peak Jitter [ps]	6.17	-
RMS Jitter [ps]	2.00	-
Power [μW]	180	629

Fig. 10 shows simulation results of the PRBS generator and checker. As shown in Fig. 10(a), each parallel PRBS-7 pattern output of the PRBS generator is shifted by 1/8 of the period of the PRBS-7 pattern. The PRBS generator was verified to operate up to 2.5 Gb/s, and its output eye diagram is shown in Fig. 10(b). The eye width, peak-to-peak jitter, and rms jitter are 393.83 ps, 6.17 ps, and 2.00 ps, respectively. To verify the PRBS checker at 4 Gb/s, 8 Gb/s output is generated using 1-Gb/s 8-bit PRBS data and transmitted to the 1:2 deserializer. Fig. 10(c) shows the simulation result of the PRBS checker with bit error counter. The bit error is accurately counted before the value of the data counter reaches its maximum, and after reaching maximum value of the data counter, the counted error value is hold.

Table I shows the performance summary table of the PRBS generator and checker.

IV. CONCLUSION

In this paper, a parallel PRBS generator and checker was proposed. The parallel PRBS generator generates 8-bit PRBS-7 data patterns using transmission gate-based D-FF. The PRBS checker can accurately detect the BER during one cycle of PRBS-7 data pattern using pattern synchronization. This design was implemented in a 65-nm CMOS process, and the PRBS generator and checker occupy an area of 75×15 μm^2 and 75×45 μm^2, respectively. It was verified that the PRBS generator operated up to 2.5 Gb/s and the PRBS checker accurately counted the bit error at 4 Gb/s.

ACKNOWLEDGMENT

This work was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant (No.2022-0-01013 & No.2022-0-01037) and in part by the National Research Foundation of Korea (NRF) grant (No.2021R1G1A1092738) funded by Korea government (MSIT). The chip fabrication and EDA tool were supported by the IC Design Education Center, Korea.

REFERENCES

[1] H. Ko, "High-speed serial link trend and technical challenge," in *IEEE Asian Solid-State Circuits Conference (ASSCC)*, Nov. 2022, pp. 1–3.
[2] K.S. Oh and X.C. Yuan, High-speed signaling: jitter modeling analysis and budgeting, Prentice Hall, 2012.
[3] M. Sakare, "A power and area efficient architecture of a PRBS Generator with Multiple Outputs," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 8, pp. 927–931, Aug. 2017.
[4] S. Kim, M. Kapur, M. Meghelli, A. Rylyakov, Y. Kwark, and D. Friedman, "45-Gb/s SiGe BiCMOS PRBS generator and PRBS checker [pseudorandom bit sequence]," in *IEEE Custom Integrated Circuits Conference (CICC)*, Sept. 2003, pp. 313–316.
[5] R. R. R. Bodha, S. Sarafi, A. Kale, M. Koberle, and J. Sturm, "A half-rate built-in self-test for high-speed serial interface using a PRBS generator and checker," in *2019 Austrochip Workshop on Microelectronics (Austrochip)*, Oct. 2019, pp. 43–46.
[6] K. Zhu and V. Saxena, "From design to test: a high-speed PRBS," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 10, pp. 2099–2107, Oct. 2018.
[7] J. Hu, Z. Zhang, and Q. Pan, "A 15-Gb/s 0.0037-mm² 0.019-pJ/bit full-rate programmable multi-pattern pseudo-random binary sequence generator," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 9, pp. 1499–1503, Sept. 2020.
[8] K. H. Kang and W. Y. Jung, "An evaluation and comparison of state-of-the-art flip-flops for low-power applications," *Journal of Integrated Circuits and Systems (JICAS)*, vol. 9, no. 2, pp. 1–5, Apr. 2023.

Jun-Cheol Lee is currently pursuing his B.S. degree at the department of Electrical Engineering, Kwangwoon University, Seoul, Korea.

His research interest includes High-Speed Interface.

Joo-Hyung Chae received his B.S. and Ph.D. degrees in Electrical Engineering from Seoul National University, Seoul, South Korea, in 2012 and 2019, respectively.

In 2013, he joined SK hynix, Icheon, South Korea, as an intern at the Department of LPDDR Memory Design. From 2019 to 2021, he was with SK hynix, Icheon, South Korea, where his work focused on GDDR memory design. In 2021, he joined Kwangwoon University, Seoul, South Korea, where he is currently an Assistant Professor of Electronics and Communications Engineering.

His research interests include the design of high-speed and low-power I/O circuits, clocking circuits, memory interfaces, and mixed-signal in-memory computing.

Dr. Chae received the Doyeon Academic Paper Award from the Inter-University Semiconductor Center (ISRC), Seoul National University, in 2020.