INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
VOLUME: 04 ISSUE: 07 | JULY – 2025 DOI: 10.55041/ISJEM04780
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

Design of High Speed UART Protocol with CRC Error Detection at IP Level

S. Mahammad Fayaz
M.Tech Student,
Department of ECE,
JNTUA CEA, Anantapur,
Andhra Pradesh, India.

Mr. K. Srinivasa Rao M.Tech, (Ph.D)
Assistant Professor
Department of ECE,
JNTUA CEA, Anantapur,
Andhra Pradesh, India

***Abstract*—In this project, we present the design and implementation of a Custom High-Speed UART Protocol with CRC Error Detection at the IP level, specifically tailored for applications that demand reliable and high-speed data transmission, such as medical imaging systems. Traditional UART protocols utilize parity bits for error detection, which are limited in their ability to detect multi-bit or burst errors, thus compromising data integrity in noise-prone environments. To overcome this limitation, the proposed design replaces parity-based error detection with CRC (Cyclic Redundancy Check) logic, enabling more robust and accurate error identification. The system is designed with a modular architecture, consisting of a UART Transmitter and Receiver integrated with CRC Generator and Checker modules. The transmission process is controlled by a finite state machine that includes framing with start, data, CRC, and stop bits. The protocol supports 8-bit data width, and CRC-16 configurations, and operates at a high baud rate of 16 Mbps with a system clock of 256 MHz, ensuring compatibility with high-throughput applications. The design is described in Verilog HDL, simulated using Xilinx Vivado, and verified through waveform analysis. The simulation results confirm correct data transmission and reception with CRC validation, and demonstrate the system's ability to detect and flag data corruption. This work enhances the reliability of UART communication and establishes a foundation for integrating error-resilient UART interfaces into modern SoC platforms.**

***Keywords*— High Speed UART, CRC Error Detection, CRC-16, Transmitter, Receiver, CRC Generator and Checker, Medical imaging, IP Core**

I. INTRODUCTION

In today's embedded systems and System-on-Chip (SoC) architectures, serial communication protocols play a crucial role in enabling data transfer between various digital modules and external devices. Among these protocols, Universal Asynchronous Receiver Transmitter (UART) remains one of the most widely adopted standards due to its simplicity, cost-effectiveness, and ease of implementation. However, as applications evolve towards high-speed and error-sensitive domains such as medical imaging, the limitations of conventional UART particularly its basic parity-bit error detection become significant. Traditional UART designs utilize single-bit parity checks that are insufficient for detecting multi-bit or burst errors, which can compromise data reliability. In environments where data integrity is paramount such as in medical diagnostic systems any undetected transmission error can lead to inaccurate results, impacting system performance and safety. To address this challenge, there is a growing need for a more robust error detection mechanism within the UART protocol This project proposes a novel design of a Custom High-Speed UART Protocol with integrated CRC (Cyclic Redundancy Check) Error Detection, implemented at the IP level using Verilog HDL and simulated on Xilinx Vivado. The design introduces CRC-16 logic to replace parity checking, significantly enhancing the protocol's ability to detect complex error patterns during serial communication. The system supports a baud rate of 16 Mbps and operates on a 256 MHz clock, making it suitable for high-throughput and real-time applications. The architecture comprises a UART Transmitter and Receiver, along with CRC Generator and Checker modules. A state machine-based control mechanism governs the transmission process, framing each data packet with a start bit, data byte, CRC bits, and stop bit. At the receiver end, the data is validated by recalculating the CRC and comparing it with the received CRC to ensure error-free reception. The entire design was functionally verified through simulation in Xilinx Vivado, using testbenches that validate correct data transmission, CRC appending and checking.

II. LITERATURE REVIEW

Several research efforts have explored the enhancement of UART protocols to meet the increasing demand for high-speed and reliable serial communication in embedded systems. This section reviews existing works and highlights the motivations for designing a custom high-speed UART protocol with CRC-based error detection. In the IEEE paper titled "Design and Implementation of High-Speed Universal Asynchronous Receiver and Transmitter (UART)", the authors propose a UART design capable of operating at higher baud rates with reduced latency by optimizing state machines and using pipeline techniques.

However, the paper still uses traditional parity-based error checking, which limits its ability to detect multiple or burst-bit errors[1]. Another related work, "Design of UART Communication System Based on Adaptive Baud Rate Technology", explores dynamic baud rate selection to improve data synchronization between devices. While this approach enhances communication flexibility, it does not address the reliability concern in error-prone environments, as it still relies on limited error detection methods[2]. In[3] a UART with

Hamming code for error correction is introduced, aiming to correct single-bit errors during data transmission. While Hamming codes add error correction capabilities, they also introduce higher overhead and complexity in hardware, making them less optimal for lightweight or power-constrained systems. Some studies have integrated CRC (Cyclic Redundancy Check) into UART designs to improve error detection without significantly increasing complexity. CRC is

known for its robustness in detecting burst errors and is widely used in high-integrity communication protocols like USB, Ethernet, and CAN. However, most UART-CRC integrations are either used in custom ASICs or not implemented as reusable IP modules suitable for SoC designs. Thus, there exists a gap in literature for a modular, IP-level UART protocol design that is high-speed, low-complexity, and capable of robust error detection using CRC logic. Addressing this gap, the proposed project aims to design and simulate a Custom High-Speed UART Protocol with CRC Error Detection, implemented in Verilog HDL, simulated in Xilinx Vivado, and tailored for integration into medical imaging SoCs.

III. EXISTING METHODOLOGY

A. Architecture of Existing UART Transmitter and Receiver

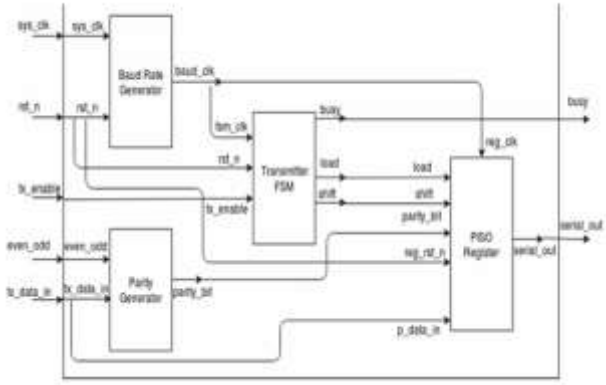

Fig 1 UART Transmitter

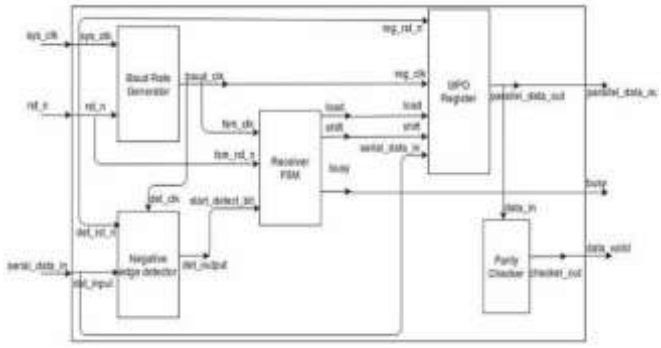

Fig 2 UART Receiver

The Universal Asynchronous Receiver Transmitter (UART) is a widely adopted serial communication protocol used in many embedded systems. The standard UART architecture is simple and effective for low- to medium-speed communication but has limitations in terms of error detection and adaptability.

B. The standard UART transmitter comprises several key functional blocks, as shown in Fig 1. These include:

a) Baud Rate Generator: Divides the input system clock to generate a specific timing signal required for serial data transmission. This timing ensures synchronization between transmitter and receiver.

b) Parity Generator: Calculates a parity bit (even or odd) based on the data being transmitted. It appends this parity bit to detect simple single-bit errors during communication.

c) Transmitter Finite State Machine (FSM): Controls the data flow during transmission. It manages insertion of start bits, data bits, parity bit, and stop bits as per UART protocol.

d) Parallel-In Serial-Out (PISO) Register: Takes parallel data input and converts it into a serial bit stream for transmission over the serial TX line.

This conventional architecture is simple and sufficient for basic data communication in low-speed systems but does not support advanced error detection mechanisms or dynamic frame configurations.

C. The conventional UART receiver architecture, illustrated in Fig 2. includes the following modules:

a) Negative Edge Detector: Detects the falling edge of the start bit, which indicates the beginning of a new data frame.

b) Baud Rate Generator: Generates the timing required for mid-bit sampling of incoming serial data.

c) Serial-In Parallel-Out (SIPO) Register: Collects the incoming serial bits and converts them into parallel format for further processing.

d) Receiver Finite State Machine (FSM): Controls the reception process, including sampling data bits at the correct baud interval, handling start/stop bit detection, and flagging when a complete frame is received.

e) Parity Checker*:* Checks the received parity bit against the calculated one to detect transmission errors.

While this setup supports fundamental error checking through parity, it is insufficient for detecting multi-bit or burst errors, which are common in noisy or high-speed communication environments such as medical imaging. Traditional UART communication protocols consist of a start bit, followed by data bits, an optional parity bit, and one or more stop bits. The parity bit is used for simple error detection by checking whether the number of ones in the data word is even or odd

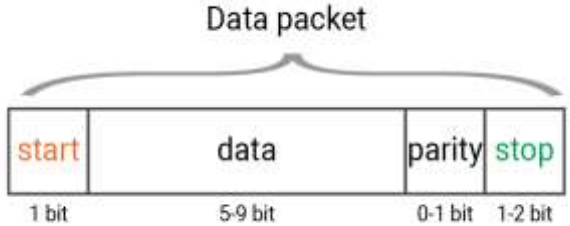

Fig.3. UART Data Frame

Traditional UART communication protocols consist of a start bit, followed by data bits, an optional parity bit, and one or more stop bits. The parity bit is used for simple error detection by checking whether the number of ones in the data word is even or odd. While this method adds minimal overhead, it is only capable of detecting single-bit errors. In cases of multiple-bit errors or burst noise, the parity check fails to detect inconsistencies, thereby compromising data integrity This limitation makes traditional UART unsuitable for applications that demand high reliability, such as medical imaging, where data corruption can have critical consequences. The figure below illustrates how parity-based error detection can fail to identify multi-bit errors, thereby motivating the need for more robust mechanisms like CRC. In conventional UART protocols, parity checking is the most common form of error detection. It is a simple mechanism that adds a single parity bit to each data frame to check for single-bit transmission errors.

D. Key Challenges and Limitations in Existing Methodology

Although conventional UART is simple and resource-efficient, it exhibits several drawbacks when applied to modern high-speed and error-sensitive applications like medical imaging:

1. Weak Error Detection: Parity checking only detects single-bit errors and fails in case of multiple or burst errors

. 2. Limited Speed: UART typically supports baud rates below 1 Mbps, unsuitable for real-time medical data.

3. Not suitable for critical applications: In domains such as medical imaging applications

IV. PROPOSED METHODOLOGY

To address the limitations of the conventional UART architecture, this project proposes a custom high-speed UART protocol with integrated CRC error detection at the IP level. The system is specifically designed for high-accuracy applications such as medical imaging, where both data integrity and throughput are critical.

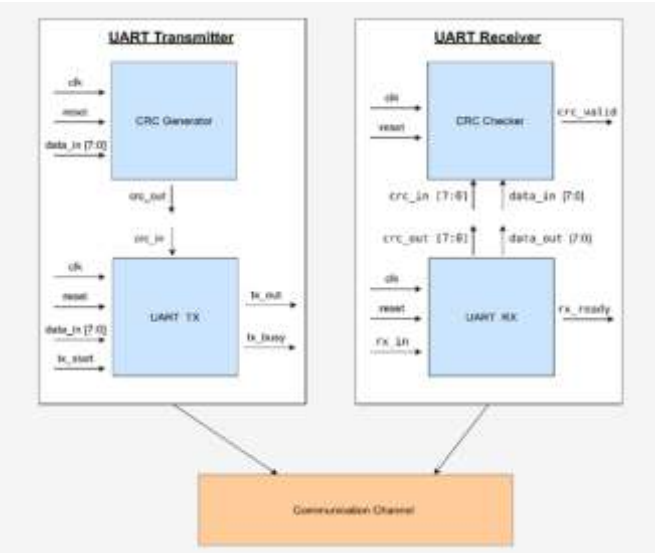

Fig 5.Proposed Architecture of High Speed UART

The proposed architecture includes four main modules:

1. CRC Generator: The CRC Generator is implemented in the transmitter side. It takes the parallel input data and computes a 16-bit CRC using a predefined polynomial (e.g., x^16 + x^15 + x^2 + 1). The generated CRC is then appended to the data frame before transmission.

2. UART Transmitter**:** The UART Transmitter is responsible for sending the combined data and CRC serially. It consists of:

- PISO Register: Converts the parallel data+CRC into serial format.
- Transmitter FSM: Controls the start bit, data bits, CRC bits, and stop bit transmission.
- Baud Rate Generator: Generates timing pulses based on a divisor, supporting up to 16 Mbps**.**

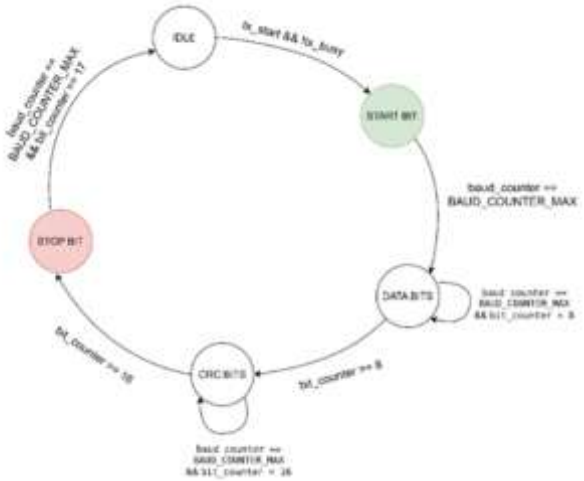

Fig 6. Transmitter FSM

3. UART Receiver: The Receiver captures the serial stream and reconstructs the original parallel data and CRC using:

- Negative Edge Detector: Detects start of transmission.
- SIPO Register: Converts serial data into parallel format.
- Receiver FSM: Synchronizes and manages incoming bits.
- Baud Rate Generator: Synchronizes sampling.

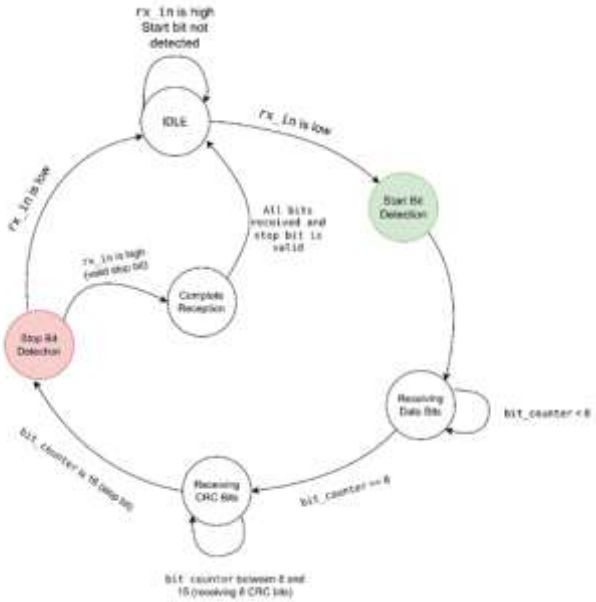

Fig 7. Receiver FSM

4. CRC Checker**:** The CRC Checker recalculates the CRC from the received data and compares it to the received CRC value.

- If the CRCs match, the crc_valid signal is set high, indicating no transmission error.
- If the CRCs mismatch, an error is detected, and crc_valid is deasserted.

V. SIMULATION RESULTS

To verify the correct functionality and reliability of the designed High-Speed UART Protocol with CRC Error Detection IP core, simulation was performed using the Xilinx vivado design suite. The simulation focuses on transmitting and receiving 8-bit serial data with CRC based error detection, ensuring data integrity across the communication channel. The waveforms illustrate the correct transmission and reception with CRC error detection of data between the UART Transmitter and Receiver modules.

A. Simulation Waveform for High Speed UART Protocol with CRC Error Detection

The simulation waveform verifies the functional behavior of the designed High Speed UART IP core as shown in fig 8

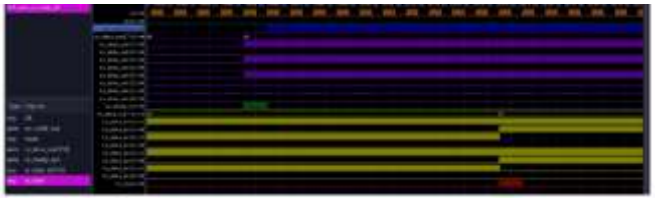

Fig 8. Proposed UART Simulation Results

The simulation waveform below represents a successful transmission and reception of data between the UART Transmitter and Receiver modules. The following observations validate the correct operation of the protocol:

- clk (Clock Signal)**:** A stable 256 MHz clock is provided to drive all synchronous elements of the design.
- reset (Active-Low Reset)**:** The system begins with an active-low reset to initialize the UART modules into a known state.
- tx_start**:** This signal is asserted high to initiate the transmission of the input data byte. It is triggered for a single clock cycle to start the transmission process.
- tx_data_in**:** The input data to be transmitted is 8'hCC (binary: 11001100). This data is processed by the CRC generator before serialization.
- rx_data_out**:** The output data received at the receiver matches the transmitted data exactly (8'hCC), indicating successful serial communication.
- rx_ready_out**:** This signal goes high for one clock cycle, indicating that valid data has been received and is available for further use.
- crc_valid_out**:** The CRC validation signal is asserted high, confirming that the CRC-16 checksum of the received data matches the transmitted CRC, indicating **no error occurred** during transmission.

The successful transmission and reception of identical data, along with the crc_valid_out signal going high, confirms the correct working of the designed UART protocol. The protocol ensures support for High Speed Serial Communication suitable for medical imaging applications and other similar data critical applications.

.

B. CRC Error Detection of High Speed UART

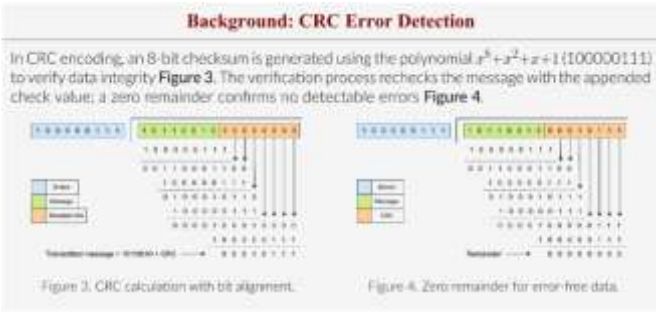

Fig 9. CRC Error Detection

The output waveform clearly demonstrates the error detection capability of the implemented CRC-16 logic in the custom high-speed UART protocol. Initially, data is transmitted from the UART Transmitter (tx_data_in = 8'hCC) upon assertion of the tx_start signal. The Receiver (rx_data_out) captures this data. The CRC logic on both the transmitting and receiving sides generates a 16-bit CRC checksum for error detection. The transmitted and received data are identical (11001100). The crc_valid_out signal is HIGH, indicating that the CRC generated at the receiver matches the CRC received with the data. This confirms that the transmission occurred without any noise or bit corruption. If the received data differs from the transmitted one then The CRC logic detects the mismatch between the calculated CRC at the receiver and the received CRC value. Consequently, the crc_valid_out signal goes LOW, indicating that the received data is corrupted. The waveform successfully validates the reliability and accuracy of the CRC-based error detection mechanism. The crc_valid_out signal serves as a robust indicator of data integrity.

C. Schematic View of High Speed UART Protocol with CRC Error Detection

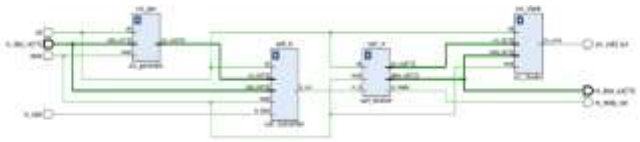

Fig 10.Schematic View of Proposed UART

Fig 10 illustrates the functional simulation schematic of the Custom High-Speed UART Protocol with CRC-16 Error Detection. This diagram, generated post-synthesis and functional verification, visually represents the signal connectivity and data flow between all major RTL components in the design. This schematic confirms the successful integration of the UART transmission/reception logic with CRC error detection and validation logic. It showcases the complete data lifecycle from input to output, highlighting both transmission accuracy and error detection reliability.

VI. CONCLUSION

This paper presents the design of a custom high-speed UART protocol integrated with CRC error detection, specifically aimed at enhancing data reliability and speed for make it suitable for medical imaging applications. Traditional UART protocols rely on parity bits, which are insufficient for

detecting complex or multiple-bit errors. By integrating a CRC generator and checker into the transmitter and receiver modules respectively, the proposed architecture significantly improves error detection capability while maintaining efficient high-speed serial communication. The complete design was modeled in Verilog HDL, functionally verified using Xilinx Vivado, and demonstrated accurate data transmission, effective CRC-based error detection. Overall, the proposed UART protocol provides a reliable and scalable IP-level solution for high-speed communication, particularly well-suited for safety-critical domains such as medical imaging applications.

VII. REFERENCES

[1] S. A. Hemalatha and K. S. Gurumurthy, "Design and Implementation of High-Speed Universal Asynchronous Receiver and Transmitter (UART)," in Proc. IEEE Int. Conf. Commun. Signal Process. (ICCSP), Chennai, India, Jul. 2020, pp. 543–547. doi: 10.1109/ICCSP48568.2020.9182271.

[2] K. Anusha and V. Balasubramanian, "Design of UART with Error Detection Mechanism," Int. J. Eng. Technol. (IJET), vol. 7, no. 2, pp. 590–593, 2015.

[3] D. Patel and S. Patel, "Design of CRC Generator and Checker for UART Communication," Int. J. Comput. Appl., vol. 179, no. 20, pp. 15–19, 2018.

[4] A. Saxena and S. Kumar, "Design of Low Power UART with Improved Error Detection," Int. J. VLSI Des. Commun. Syst. (VLSICS), vol. 6, no. 3, pp. 21–30, 2015.

[5] K. R. Lakshmi kandan and S. Lavanya, "CRC Based Error Detection for Serial Communication," Int. Res. J. Eng. Technol. (IRJET), vol. 6, no. 4, pp. 1742–1746, 2019.

[6] H. Park and B. K. Kim, "A Reliable Data Communication Protocol for Medical Imaging Applications," IEEE Trans. Consum. Electron., vol. 58, no. 4, pp. 1121–1128, Nov. 2012.

[7] J. F. Wakerly, Digital Design: Principles and Practices, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2005.

[8] Xilinx Inc., "AXI UART Lite v2.0," Product Guide PG142, 2021. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/axi_uartlite/v2_0/pg142-axi-uartlite.pdf

[9] D. A. Patterson and J. L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, 5th ed. San Francisco, CA, USA: Morgan Kaufmann, 2014.

[10] W. Stallings, Data and Computer Communications, 10th ed. Boston, MA, USA: Pearson, 2013.

[11] J. Song and D. Wang, "Design of Reliable Serial Communication Based on UART and CRC," in Proc. Int. Conf. Electron. Commun. Control (ICECC), Ningbo, China, Sep. 2011, pp. 198–201. doi: 10.1109/ICECC.2011.6067495.

[12] B. W. Johnson, "CRC and Checksums for Embedded Systems," IEEE Potentials, vol. 18, no. 4, pp. 4–7, Oct. 1999.

[13] M. S. Islam and S. U. Mahmud, "High Speed UART Design Using Verilog HDL," in Proc. Int. Conf. Electr. Eng. Inf. Commun. Technol. (ICEEICT), Dhaka, Bangladesh, May 2015, pp. 1–5. doi: 10.1109/ICEEICT.2015.7307459.

[14] A. S. Mohan and K. R. Mahajan, "Design of Low-Power UART with Efficient Error Detection Technique for Embedded Systems," Int. J. Eng. Res. Technol. (IJERT), vol. 7, no. 5, pp. 98–102, 2018.

[15] D. Kalra and A. Yadav, "Implementation of UART Using VHDL," Int. J. Adv. Res. Comput. Sci. Softw. Eng., vol. 4, no. 8, pp. 267–270, 2014.

[16] S. E. Hossain, M. S. Alam, and M. H. Kabir, "Design of a Reliable UART Module Based on FPGA," in Proc. Int. Conf. Informatics, Electron. Vision (ICIEV), Fukuoka, Japan, Jun. 2012, pp. 479–483. doi: 10.1109/ICIEV.2012.6317476.

[17] T. Li, J. Wu, and W. Li, "An Efficient Hardware Implementation of CRC for High-Speed Communication," in Proc. Int. Conf. Commun. Softw. Netw. (ICCSN), Singapore, Mar. 2013, pp. 123–127. doi: 10.1109/ICCSN.2013.6562639.