ISSN 2959-6157

Development and Evaluation of a Robust UART Communication System with Enhanced Interference Resistance

Rui Li

Department of Electronic and Information Engineering, Fuzhou University, Fuzhou, 350000, China
832201108@fzu.edu.cn

Abstract:

Interference is a significant issue in UART communication, often causing data jitter and distortion, particularly in industrial environments. This study addresses these challenges by developing an anti-interference UART system that includes the design and implementation of receiving, processing, and sending modules. The system utilizes techniques such as multi-level sampling, synchronization, and cumulative decision-making to enhance data transmission accuracy and reliability. In embedded systems, electromagnetic interference (EMI) poses a major problem, leading to potential data loss and communication failures. Reliable communication is crucial for maintaining the performance and safety of industrial operations. The anti-interference UART system developed in this study aims to improve operational stability by mitigating the effects of EMI. The system's performance was validated through extensive simulations, demonstrating its ability to maintain stable and accurate data transmission even under various interference conditions. The results show significant improvements in data integrity and communication reliability. This paper presents a comprehensive approach to designing, implementing, and validating a robust UART system that can effectively operate in challenging environments, ensuring reliable data communication and enhancing overall system performance.

Keywords: UART protocol, Anti-interference, Data transmission reliability

1. Introduction

In embedded communications, a communication protocol is a set of rules that allows two or more entities of a communication system to transmit information. Universal Asynchronous Receiver/Transmitter (UART) is a common hardware communication protocol that enables serial data transmission between devices without a shared clock signal. It is a critical component in embedded systems, specialized computing systems designed for specific control functions within larger devices, such as automotive controls and industrial machines. Embedded systems rely on communication protocols like UART, SPI, and I2C for effective data exchange. Key components of embedded systems include communication interfaces, which manage data transfer between devices; buffers, which temporarily store data to handle differences in processing speed; storage units, which provide space for saving instructions and data; and buses, which facilitate data transfer between the microcontroller, memory, and peripherals, ensuring smooth system performance. Reliable communication interfaces are essential in embedded systems to maintain data integrity and ensure system reliability.

In the real industrial environment, electromagnetic interference (EMI) is a major problem in the communication process. Once the communication process is affected by EMI, it may lead to data jitter and even distortion. However, in embedded communication systems, especially those using the UART protocol, there are significant challenges in maintaining data stability due to various types of interference. These interferences can be roughly divided into external interference and internal interference. External interference includes EMI generated by electrical, mechanical, and other electronic devices, which produces noise that interferes with signal integrity, crosstalk when signals from adjacent communication channels interfere with each other, and signal attenuation, which is a reduction in signal strength with distance. Internal disturbances include thermal noise, random electrical noise generated by the thermal agitation of electrons in the communication medium, and power supply changes that introduce noise and affect the performance of the communication circuit.

To solve these problems, this study aims to improve the reliability of the UART system by designing and implementing an anti-jamming UART receiving module (named ‚uart_anti_interference_rx'), which can accurately receive serial data and convert it to parallel data. In addition, data processing modules will be developed to meet specific application requirements, converting 8-bit data to bn* 8-bit

data, and a reliable UART sending module (named ‚UART tx‘) will be designed to ensure stable data transmission. These improvements are designed to ensure accurate and reliable communication in industrial environments, improving operational stability and safety.

This paper reviews improvements in UART and SPI protocols for enhanced transmission efficiency and anti-interference. It then describes the design and implementation of the UART system's main modules and their interactions. Following this, the integration of these system modules and their application in industrial automation, highlighting their enhanced anti-interference capabilities, are detailed. The paper also presents the experimental setup, simulations, and performance evaluation of the UART system. Finally, it summarizes the findings, and contributions to the field, and suggests future work to further improve the system's anti-interference capabilities.

2. Literature Review

At present, in embedded systems, the research on communication protocols has accumulated a lot of relevant work. For example, the UART communication protocol of embedded system in industrial applications is optimized by frequency multiplication sampling and asynchronous FIFO, which improves transmission efficiency, bit error rate and anti-interference ability [1], the SPI communication protocol between microprocessors in engineering applications improves accuracy in complex environments and achieves data reliability by using closed-loop diagnostics and retransmission strategies [2], the low-power UART communication protocol is implemented on 45nm FPGas using the frequency scaling and High Speed Transceiver Logic (HSTL) standard, enabling efficient data transfer for microcontroller-based systems with reduced power consumption [3], the interface between MEMS motion sensor and FPGA based on I2C protocol can realize the efficient transmission and denoising of sensor data in motion detection applications [4], a high-speed UART based on the RS-422 standard and FPGA built-in self-check (BIST) function achieves reliable data transmission with low bit error rate and is suitable for complex integrated circuits [5]. From this, it can be seen that the optimization of the UART and SPI protocols improves transmission efficiency, bit error rate, and anti-interference, while the low power consumption and high-speed implementation on FPGAs and the efficient I2C interface of MEMS sensors enable reliable data transmission, making these protocols suitable for a variety of applications from industrial environments to motion detection and complex integrated circuits.

At the same time, in the embedded field, the research on interference reduction in communication systems has also made great progress. For example,improve the robustness of UART architecture by using recursive runs and filters to improve noise performance, optimize asynchronous serial communication for embedded systems, and enhance data integrity [6], the information is encrypted using a nonlinear coupling function and decrypted using dynamic Bayesian inference to ensure strong noise resistance, ensuring the robustness of the communication system even under high noise conditions [7], a decentralized authentication protocol for autonomous embedded systems using a Schnorr-based multi-signature scheme enhances robustness by withstanding network outages and arbitrary device outages [8], a random communication protocol is applied to improve the accuracy and reliability of fault detection, enabling robust fault detection in uncertain delay systems with measured outliers [9], and through the SDN flat communication protocol architecture and high reliability guarantee mechanism, the stability and reliability of the communication process are protected [10]. Therefore, it is concluded that enhancing robustness or anti-interference includes various methods to improve noise performance, ensure strong anti-interference, and maintain reliable communication and fault detection. Each method has its advantages, but the complexity and implementation methods vary.

3. Methods and Modeling Techniques

As for the overall architecture of the communication system in this study(as shown in Figure 1), the system includes the following main modules: The control unit is responsible for coordinating and managing the operation of the entire UART communication system, receiving control signals from the peripheral bus, and sending control signals to the receiver and transmitter modules. The baud rate generator generates a clock signal matching the set baud rate, and receives the clock signal from the clock source, The generated clock signal is sent to the receive shifter and transmit shifter to ensure the synchronization of data transmission. The receiver module includes the receive buffer register, the receive holding register (used in non-FIFO mode only) and receive FIFO register (used in FIFO mode). The receiving shifter converts the received serial data into parallel data and stores it through the receiving buffer register. The received data is managed by the control unit to ensure data integrity and correctness. The transmitter module includes the transmit buffer register, the transmit holding register (used in non-FIFO mode only) and transmit FIFO register (used in FIFO mode). The transmission shifter converts parallel data to serial data and sends it out via TXDn. The transmission process is controlled by the control unit to ensure the accurate transmission of data. The whole system realizes efficient

and reliable data transmission through the cooperative work of the control unit, baud rate generator, receiving module, and sending module.

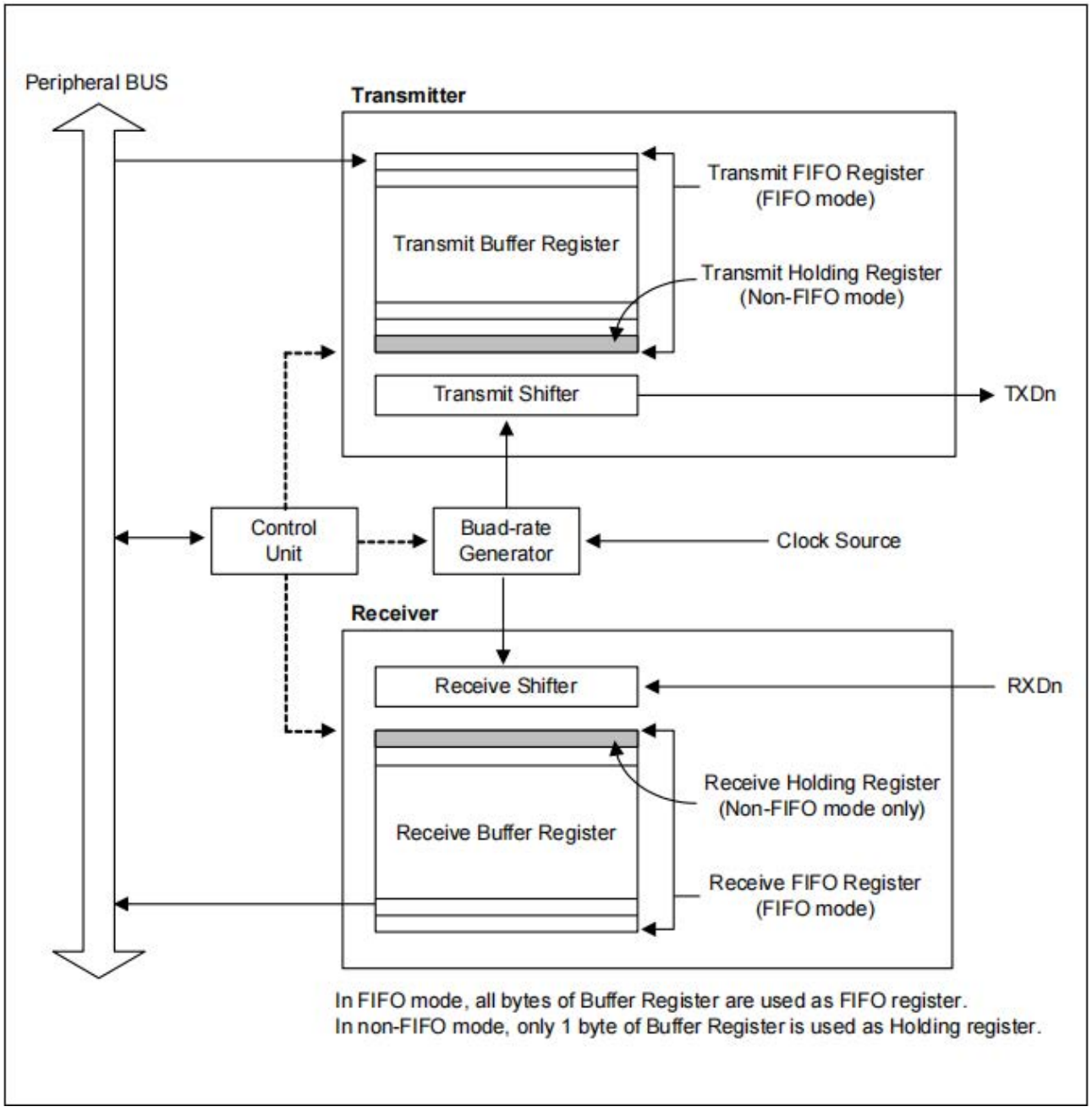

Fig 1. Overall Architecture of the UART Communication System[11]

For the specific analysis of the above overall framework, the specific institutions applied are shown in Figure 2. It consists of an Upper Computer (Upper Computer), a receiving terminal (rx terminal), a Main Module (Data Processing), and a sending terminal (uart_tx5 terminal). The host computer, as the Master, is responsible for sending and receiving data. The sending signals include the rx signal (sending to the receiving terminal), and the receiving signals include bitout (receiving from the sending terminal) and tx_busy (receiving from the sending terminal). The receiving terminal is responsible for receiving the serial data sent by the host computer and converting it into parallel data, with input signals including rx signal (from the host computer), rst (reset signal) and clk (clock signal), and output signals including [7:0] outdata (parallel data, Send to the main module) and rx_done (receive completion signal, send to the main module). The main module is responsible for processing the received data, such as

converting 8-bit data to bn* 8-bit data, and the input signals include [7:0] outdata (from the receiving terminal), rx_done (received completion signal, from the receiving terminal), rst (reset signal) and clk (clock signal). The output signals include [bn*8-1:0] din (processed data, sent to the sending terminal) and en (enable signal, sent to the sending terminal). The sending terminal is responsible for converting the parallel data processed by the main module into serial data and sending it to the host computer. The input signals include [bn*8-1:0] din (to the autonomous module), en (enable signal, to the autonomous module), rst (reset signal), and clk (clock signal). The output signals include bitout (sent to the host computer) and tx_busy (busy signal, sent to the host computer). In the process of data received, the host computer sends serial data to the receiving terminal through rx signal, the receiving terminal converts the serial data into parallel data [7:0] outdata and informs the main module of the completion of data receiving through the signal rx_done. The main module receives parallel data [7:0] outdata and rx_done signals for data processing, such as converting 8-bit data into bn* 8-bit data, and sending the processed data to the sending terminal via signals [bn*8-1:0] din and en. The sending terminal receives the processed parallel data, converts it into serial data through the sending shifter, sends it back to the host computer through signal bitout, and indicates the data transmission status through signal tx_busy. In the figure, the arrows between the modules indicate the direction and type of signal transmission. For example, rx signal indicates that the serial data sent by the host computer is transmitted to the receiving terminal; [7:0] outdata indicates that the parallel data output by the receiving terminal is transmitted to the main module; rx_done indicates that the receiving terminal informs the main module of the completion of data reception. [bn*8-1:0] din and en indicate that the processed data output by the main module and the enable signal are transmitted to the sending terminal; bitout indicates that the serial data sent by the sending terminal is transmitted back to the host computer; tx_busy indicates that the busy signal of the sending terminal informs the host computer of the current sending status. Each module also includes the necessary registers, such as synchronization register, accumulation register and sampling counter in the receiving terminal, buffer register and shift register in the sending terminal, to ensure the accurate reception and transmission of data. Through the cooperative work of the above modules, the system realizes reliable anti-interference UART communication and ensures the stability and reliability of data transmission. In addition, the frame format and transmission rate of the UART protocol also played a key role in the design. In this experiment, the sampling clock is 50MHz and the baud rate is 115200. The transmission process includes a start bit (low level), eight data bits (from low to high level), one optional check bit, and one or more stop bits (high level), thus ensuring the accuracy and reliability of the data transmission.

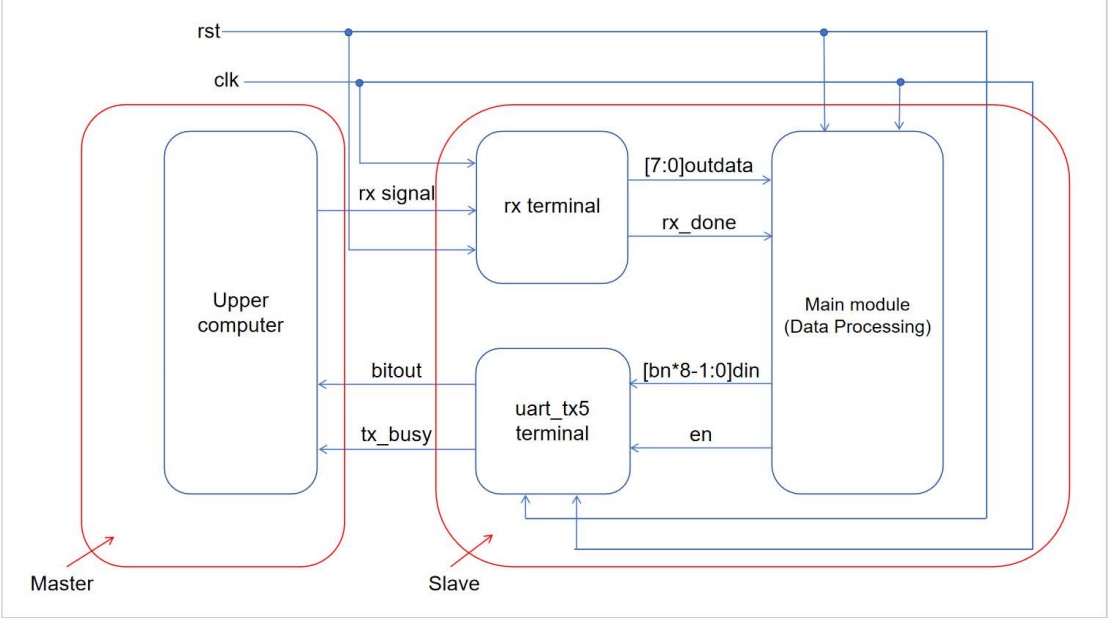

Fig 2. Block Diagram of the UART System Components and Signal Flow

In addition, to enhance the anti-interference ability of the UART receiving module, this experiment adopts a variety of technical measures to ensure the stability of the system in the industrial environment. First, the receiving module

synchronizes the input signal with the synchronization register to eliminate metastability and reduce the influence of electromagnetic interference by sampling multiple times. When the baud rate is 115200, each data is sampled 16 times, and the middle 6 times are taken as valid samples, and the level of data bit is determined according to the sampling results, which reduces the possibility of data jitter. In addition, the receiving module verifies the validity of the starting bit and the data bit through the cumulative decision method to ensure the accuracy of the received data. The module also introduces reset and enable signals to reset the status when a reset signal is detected, the reception is complete, or the start bit fails to maintain the reliable operation of the system. Through these technologies, the receiving module significantly enhances the anti-interference capability, ensuring the stability and reliability of the UART communication system in the industrial environment.

4. Applications and Integrations

In the above system, the receiving, processing, and sending modules are tightly integrated to form a complete UART communication system. The receiving module of the system first receives the serial data sent by the host computer and uses synchronous register and multiple sampling technology to enhance the anti-interference ability. Once the reception is complete, the data is converted into a parallel format and transmitted to the main module for processing. The main module is responsible for converting the received data into data suitable for transmission format and generating the corresponding enable signal. The processed data and the enable signal are transmitted to the sending module, which converts the parallel data back to serial data and sends the data back to the host computer by sending signals. In the whole process of data transmission, modules communicate and cooperate through clear signal interfaces. For example, rx signal represents serial data from the host computer to the receiving module, [7:0] outdata represents parallel data output by the receiving module, and rx_done represents data receiving completion signal. [bn*8-1:0] din and en represent the data and enable signal after processing by the main module, bitout represents the serial data sent by the sending module, and tx_busy represents the busy signal during transmission. Through this modular integrated design, the system not only improves the stability of data transmission but also enhances the anti-interference ability effectively.

The design and implementation of the anti-interference UART system make it have significant advantages in many practical application scenarios, especially in industrial automation control, the anti-interference UART system can be used for data communication between PLC (programmable logic controller) and various sensors and actuators. The specific application steps include: First, various sensors (such as temperature sensors, pressure sensors, photoelectric sensors.) are connected to the system through the UART interface, and these sensors are responsible for collecting various data in the industrial environment, such as temperature, pressure, and light intensity. The sensor then sends the collected data to the receiving module in serial mode through the UART interface. The receiving module performs preliminary processing of the data sent by the sensor, converts the serial data into parallel data, and transmits it to the main module for further processing. The main module checks and processes the received data, such as filtering noise, converting data format, and generates corresponding control signals and data. The processed data and control signals are transmitted to the sending module, which re-converts the parallel data into serial data and sends it back to the PLC via the UART interface. The PLC performs corresponding operations according to the received data and control signals, such as starting or stopping the actuator, adjusting production parameters. In the whole process, the anti-interference UART system ensures the accuracy and timeliness of the data through reliable data transmission and processing, effectively reduces the impact of electromagnetic interference on data transmission in the industrial environment and improves the stability and reliability of the system. For example, in a practical application scenario, the temperature sensor sends the temperature data to the PLC through the UART interface, and the PLC controls the start and stop of the cooling system according to the received temperature data, thus realizing the automation and intelligence of the industrial production process. This application scenario not only shows the practical application of anti-interference technology in industrial automation but also reflects the collaborative work and efficient integration of various modules in the system design.

5. Experiments and Model Evaluation

To verify the anti-interference ability and overall performance of the UART system, hardware description language code written by VHDL/Verilog, modelsim simulation tool, and data analysis software are used in the software part of this experiment. The experimental process includes the construction of the overall architecture of the system, the verification of the functions of each module, and the overall performance test after the integration of the system, to ensure that the system can work stably under various interference environments, and that there are no obvious errors in the process of data receiving and

sending.

The simulation results of each module are analyzed as follows:

First, the input signal rx is synchronized with multiple registers to eliminate metastability. At the rising edge of the clock, the rx signal is captured and transmitted step by step, and the falling edge of the signal is detected by comparing the output of the last two stages of the register, which is used to determine the starting bit(Figure 3). When a failure of the start bit is detected, 'rx_state' is pulled down. For example, if the start bit is high (while the start bit should be low) more than 2 times during the sampling process when the count reaches 12, the start bit may be invalid (affected by interference or noise), at which point 'rx_state' is pulled down(Figure 4). After the start bit is valid, the receiving status is raised. The counter counts at the baud rate, resets when it reaches a set value, and generates a sampling flag in the middle. Another counter records the number of samples taken and resets when a certain value is reached or an error is detected. The input signal is sampled several times according to the sampling mark, and the result is accumulated. Sample each data bit 16 times, and take the middle 6 times to ensure stability. When the sampling counter reaches 160, the result is assigned to the output data, a receive complete signal is generated, and the receive state is pulled down(-Figure 5).

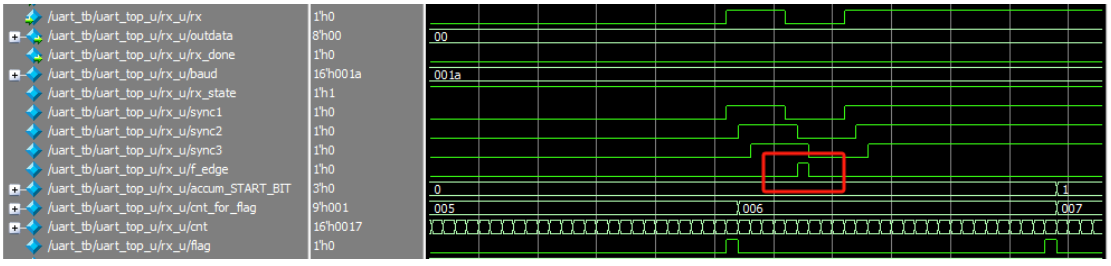

Fig 3. Synchronization and Edge Detection for Start Bit

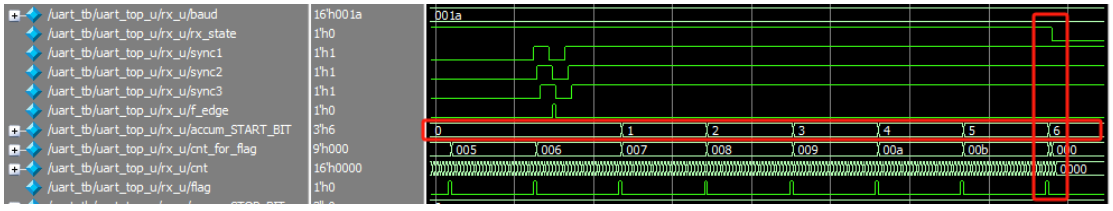

Fig 4. Invalid Start Bit Detection Due to High-Level Interference

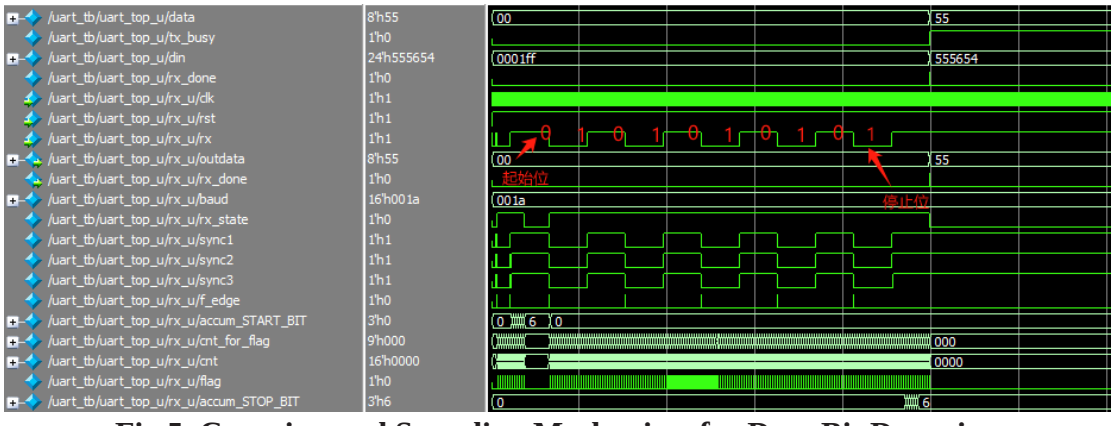

Fig 5. Counting and Sampling Mechanism for Data Bit Detection

After the data is sent out to the receiving module, the data enters the main module, processes the accepted 8-bit data, converts it into bn* 8-bit data, and upload it to the host computer, where the value of bn is 3(Figure 6).

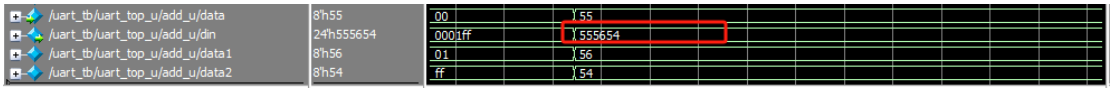

Fig 6. 8-bit Data into bn*8-bit Data in the Main Module

After the data enters the sending module, initialize the submodule. When the rising edge of the received signal is detected, the enable signal is pulled higher and data transmission begins (Figure 7); When the receiving status is lowered, the sending busy signal indicates transmission. Transmits the minimum eight bits of data to the submodule and sends enable and busy signals. Synchronize the busy signal, detect the falling edge, and mark the completion of eight-bit data transmission. Then the next eight bits of data are stored, and the counting signal increases. After the count reaches the set value, data transmission is completed, registers and signals are cleared, and the sending module enters the idle state (Figure 8).

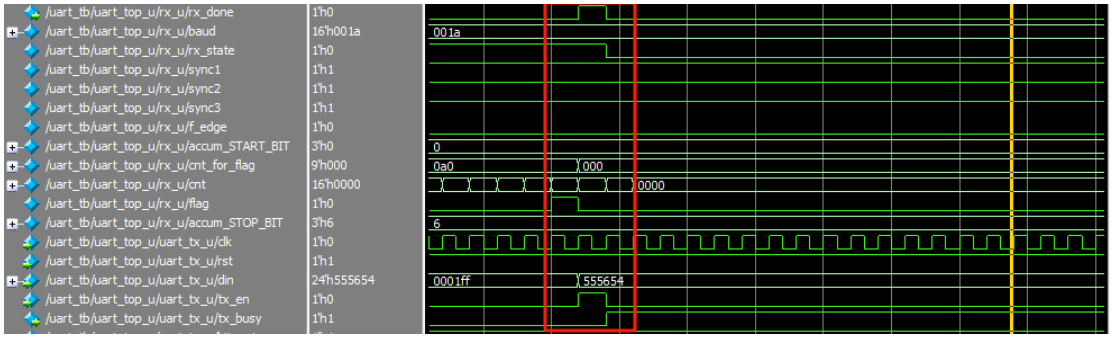

Fig 7. Initialization and Data Transmission in the UART Transmitting Module

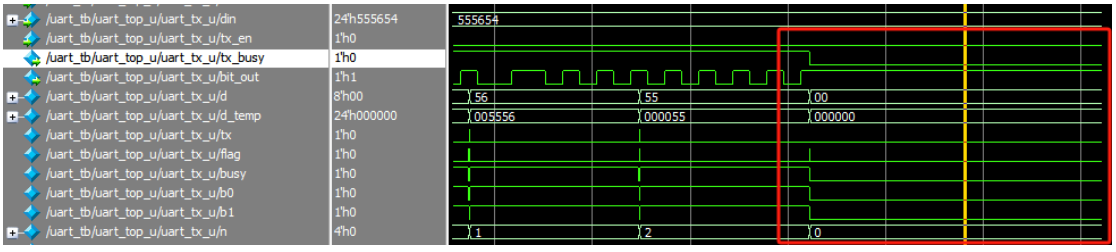

Fig. 8 Busy Signal Synchronization and Data Transmission Completion

At the same time, after the data enters the submodule, the clock counter starts counting at the baud rate, resets when it reaches the set value, and adds one to the send counter. When the send counter reaches 9 and the clock counter reaches the set value, the send flag and busy signal pull down, enter the idle state, and wait for the next data transmission. During this period, according to the value of the send counter, the corresponding data bit is assigned (0 is the start bit, 9 is the stop bit, and the middle value is sampled)(Figure 9).

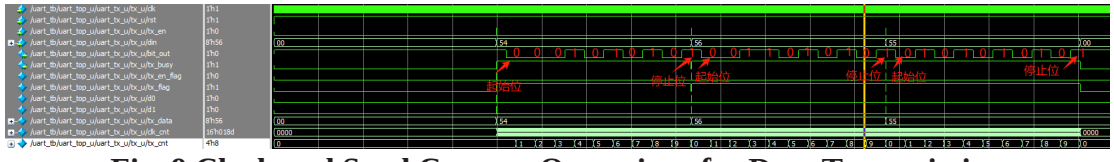

Fig. 9 Clock and Send Counter Operations for Data Transmission

Since then, the performance of the system is also evaluated comprehensively and compared with traditional non-anti-interference UART systems. The experimental results show that the anti-interference UART system can work stably under various interference environments, and there is no obvious error in the process of data receiving and sending, which shows high reliability. Through simulation and actual testing, the accuracy of the system in data processing and transmission is verified to ensure that the received data is consistent with the sent data, thus ensuring data integrity. The test in the electromagnetic interference environment further verifies the anti-interference performance of the system. The results show that the anti-interference UART system can effectively resist electromagnetic interference and maintain the stability and reliability of data transmission. Compared with the traditional non-anti-interference UART system, the system has a significant improvement in anti-interference ability, data transmission stability, and reliability. These improvements give the system a wide application prospect in industrial automation, sensor networks, medical equipment, and other applications.

6. Conclusion

This research designs and implements a UART communication system with anti-interference ability, including

receiving module, data processing module, and sending module. Through experiments and simulations, the modules can work stably in various interference environments, and there is no obvious error in the process of data receiving and sending, showing high reliability. The receiving module can effectively reduce the influence of electromagnetic interference on data transmission and ensure the accuracy of data through multiple sampling and cumulative judgment. The data processing module can accurately convert the received 8-bit data into the required format, while the sending module ensures the integrity of the data during transmission. The overall system has been significantly improved in anti-interference ability, data transmission stability, and reliability.

At the same time, this research also makes a certain contribution to the anti-interference design of UART communication system. By introducing the techniques of multiple sampling, synchronous register, and summation decision, the anti-interference ability of the system is enhanced effectively, and the accuracy and reliability of data transmission are ensured. These technologies are not only applicable to industrial automation control, sensor networks, and medical devices but also provide a reference for other application scenarios that require highly reliable data transmission. In addition, this study verifies the feasibility and effectiveness of the system in practical application through detailed experimental Settings and simulation analysis, which provides a solid foundation for further research and practical application.

However, although the UART communication system in this study has achieved remarkable results in anti-interference performance, there is still room for further improvement. Future research can focus on the following aspects: First, more advanced anti-jamming techniques, such as adaptive filtering and intelligent error correction algorithms, can be explored to further improve the reliability of the system. Secondly, consider integrating more functional modules, such as data encryption and decryption modules, to enhance the security of the system. In addition,it is also possible to try to apply the system to more practical scenarios, and further verify its performance and stability through practical tests. Finally, the system can be optimized to reduce power consumption and cost, so that it can be widely used in more embedded systems.

References

[1] W. Wang, "Optimization of UART Communication Protocol Based on Frequency Multiplier Sampling Technology and Asynchronous FIFO," 2023 IEEE 2nd International Conference on Electrical Engineering, Big Data and Algorithms (EEBDA), Changchun, China, 2023, pp. 280-285, doi: 10.1109/EEBDA56825.2023.10090630.

[2] H. Wei, J. Mei, J. Liu, Z. Wang, S. Fang and L. Wang, "Application Research on Data Reliability of SPI Bus between Microprocessors," 2021 IEEE 4th International Conference on Electronics Technology (ICET), Chengdu, China, 2021, pp. 347-351, doi: 10.1109/ICET51757.2021.9450983.

[3] A. Kumar, B. Pandey, D. M. Akbar Hussain, M. Atiqur Rahman, V. Jain and A. Bahanasse, "Frequency Scaling and High Speed Transceiver Logic Based Low Power UART design on 45nm FPGA," 2019 11th International Conference on Computational Intelligence and Communication Networks (CICN), Honolulu, HI, USA, 2019, pp. 88-92, doi: 10.1109/CICN.2019.8902375.

[4] R. S. S. Kumari and C. Gayathri, "Interfacing of MEMS motion sensor with FPGA using I2C protocol," 2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), Coimbatore, India, 2017, pp. 1-5, doi: 10.1109/ICIIECS.2017.8275932.

[5] S. Saha, M. A. Rahman and A. Thakur, "Design and implementation of a BIST embedded high speed RS-422 utilized UART over FPGA," 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 2013, pp. 1-5, doi: 10.1109/ICCCNT.2013.6726481.

[6] H. Patel, S. Trivedi, R. Neelkanthan and V. R. Gujraty, "A Robust UART Architecture Based on Recursive Running Sum Filter for Better Noise Performance," 20th International Conference on VLSI Design held jointly with 6th International Conference on Embedded Systems (VLSID'07), Bangalore, India, 2007, pp. 819-823, doi: 10.1109/VLSID.2007.25.

[7] G. Nadzinski et al., "Experimental Realization of the Coupling Function Secure Communications Protocol and Analysis of Its Noise Robustness," in IEEE Transactions on Information Forensics and Security, vol. 13, no. 10, pp. 2591-2601, Oct. 2018, doi: 10.1109/TIFS.2018.2825147.

[8] F. Kohnhäuser, N. Büscher and S. Katzenbeisser, "A Practical Attestation Protocol for Autonomous Embedded Systems," 2019 IEEE European Symposium on Security and Privacy (EuroS&P), Stockholm, Sweden, 2019, pp. 263-278, doi: 10.1109/EuroSP.2019.00028.

[9] W. Chen, J. Hu, X. Yu, D. Chen and Z. Wu, "Robust Fault Detection for Uncertain Delayed Systems With Measurement Outliers Under Stochastic Communication Protocol," in IEEE Transactions on Signal and Information Processing over Networks, vol. 8, pp. 684-701, 2022, doi: 10.1109/TSIPN.2022.3192650.

[10] Z. Wang, Y. Wang, L. Teng, Y. Hong and F. Chen, "Reliability Guarantee Mechanism and Flat Communication Protocol Structure for Power System Protection Communication Networks in Energy Internet," 2018 5th International Conference on Information Science and Control Engineering (ICISCE), Zhengzhou, China, 2018, pp. 1188-1192, doi: 10.1109/ICISCE.2018.00243.

[11] "Figure 28-1 in SEC Exynos4412 User's Manual," Samsung Electronics Co., Ltd., 2012.