International Journal of Frontiers in Engineering Technology
ISSN 2706-655X Vol.6, Issue 1: 125-130, DOI: 10.25236/IJFET.2024.060120

Research on high-speed communication mechanism of time-triggered network based on RGMII

Changhao Wang, Tiantian Li*

School of Electronic Information and Artificial Intelligence, Shaanxi Uninersity of Science & Technology, Xi'an, China
**Corresponding author*

Abstract: *With the rapid development of time-triggered network, the demand for its network performance is getting higher and higher. The transmission mechanism of time-triggered network is time-triggered mechanism, which is used to study the high-speed communication of time-triggered network. This paper proposes a high-speed communication design scheme for time-triggered network based on RGMII interface: time-triggered network transmission control module, FPGA board for time-triggered network transmission control, time-triggered encapsulation and analysis module for communication with host computer, configuration module for parameter configuration, cyclic redundancy check module for data error check. The communication between PHY layer and MAC layer is carried out by RGMII interface. Finally, simulation verification and upper board verification are carried out to realize the high-speed transmission of time-triggered network based on RGMII interface.*

Keywords: *Time-triggered network, SAE AS6802, High speed communication*

1. Introduction

Time-triggered network not only has the advantages of traditional Ethernet technology, but also has the characteristics of Time-trigger (TT) transmission mechanism[1], which has high real-time, high certainty and high fault tolerance, and establishes global clock synchronization during transmission and has strict time certainty[2].

Time-Triggered networks are studied based on time-triggered Protocol (TTP), which was proposed by Professor Kopetz of Vienna University of Technology in Austria in 1994[3]. In 2003, Professor Kopetz proposed that the architecture of time-triggered network belongs to the embedded real-time system[4], which adopts a unified global clock to ensure the time certainty of each node in the network. In 2005, Professor Kopetz pointed out that both real-time data and non-real-time data can be transmitted through time-triggered networks[5], and time-triggered networks have been widely developed in the field of real-time. In 2011, SAE Organization in the United States explained the synchronization mechanism and operation method of time-triggered networks, and formulated SAE AS6802 time synchronization protocol standard[6][7]. In the synchronization process, the clock transparency mechanism of IEEE 1588 protocol[8] was retained, and fault tolerance mechanism was added[9]. To deal with faulty nodes affected by synchronization accuracy[10]. In 2014, NASA successfully launched a manned spacecraft using the AS6802 time-synchronization protocol[11]. Some scientific research institutions began to study time-triggered networks, such as the University of Vienna and the University of Verona, among which the University of Vienna played a leading role in the study of time-triggered networks, and conducted standardized and breakthrough research on time-triggered networks[12]. In 2013, scholars from the University of Electronic Science and Technology of China studied the high accuracy of clock synchronization in time-triggered networks by using network simulation and calculation[13], which ensured the high accuracy of clocks in time-triggered networks and satisfied the real-time and reliability of data transmission. Scholars from Beijing University of Aeronautics and Astronautics pointed out that time-triggered network is an important technology for the development of civil aircraft science and technology, which can meet the needs of the latest aviation network system[11][14].

Many research institutions and colleges have successfully applied the technology of time-triggered network to aviation, automobile and other fields, and achieved relatively good results. However, the progress in the high-speed transmission and fault tolerance mechanism of time-triggered network is slow, so it is necessary to study the high-speed communication of time-triggered network with RGMII interface. By designing the high-speed communication mechanism of time-triggered network based on RGMII

ISSN 2706-655X Vol.6, Issue 1: 125-130, DOI: 10.25236/IJFET.2024.060120

interface, the time synchronization and service transmission functions of the FPGA board of the end node of the time-triggered network are realized, and the transmission rate and resource utilization of the communication service in the network are improved. To improve the reliability of high-speed communication design of time-triggered network, a simulation test environment is built to verify the high-speed communication mechanism of time-triggered network nodes and board level verification.

2. Design and implementation

2.1. Overall design scheme

In order to better study the high-speed communication mechanism of time-triggered network based on RGMII interface, it is necessary to design and implement the FPGA board at the end of the time-triggered network, carry out the design of the logic level of time-triggered high-speed communication, and complete the data frame format implementation, high-speed transmission control, cyclic redundancy check, information configuration and other functions required by the time-triggered network. The overall module distribution diagram of the high-speed communication mechanism design of time-triggered network based on RGMII interface is shown in *Figure 1*. It is necessary to implement the encapsulation module of time-triggered frame, the decapsulation module of time-triggered frame, the configuration module to input the network parameters to be configured, the cyclic redundancy check to detect the error of the valid data sent, and if the data is found to be incorrect, it will be discarded, and the implementation of the top-level transmission control module of time-triggered network based on RGMII interface. Finally, the logic simulation of each module and the board level verification of the whole design are carried out to realize the high-speed communication mechanism of the time-triggered network.

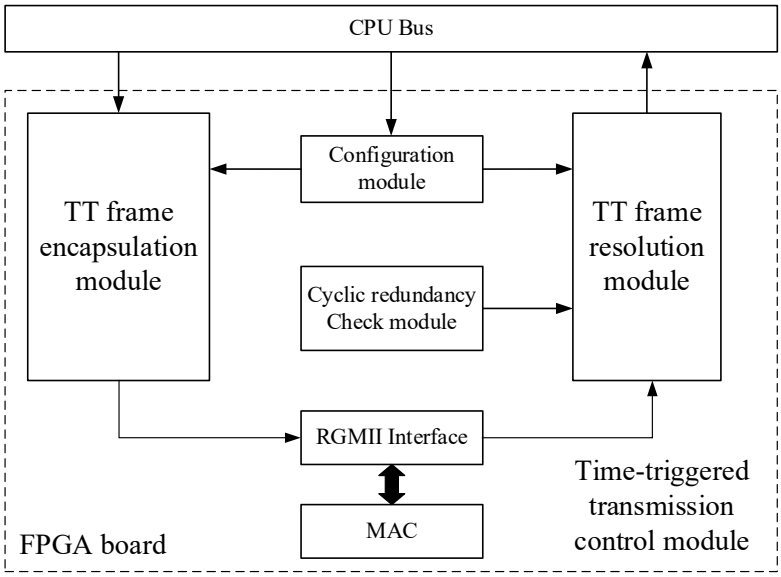

Figure 1: The overall module distribution diagram of high-speed communication mechanism design in time-triggered network.

2.2. Encapsulation and decapsulation module design and implementation

To realize the encapsulation and decapsulation of time-triggered frames is to complete the format of time-triggered data frames - the receiving and sending of time-triggered frames. The receiving process is that the Ethernet physical layer receives the incoming data and parameters from the PC side and encapsulates them into time-triggered frames. The sending process is that the Ethernet physical layer sends time-triggered frames to the PC side and decapsulates the data and parameters of the frames. The format of time-triggered frame is improved based on the standard Ethernet frame format. It is composed of eight parts, including 7-byte frame synchronization precursor code, 1-byte frame delimiter, 6-byte destination address and 6-byte source address, 2-byte time triggered frame service type with the value 0x88d7, and the minimum data payload of 46 bytes. When the number of bytes is less than 46 bytes, the serial number of the 1-byte multi-channel redundant identification and the 4-byte CRC check code are added. The virtual link number of the time-triggered frame is stored in the lower 16 bits of the destination address, and the data payload includes a 20-byte IP header, an 8-byte UDP header, and a valid data portion.

ISSN 2706-655X Vol.6, Issue 1: 125-130, DOI: 10.25236/IJFET.2024.060120

Its components are shown in *Figure 2*.

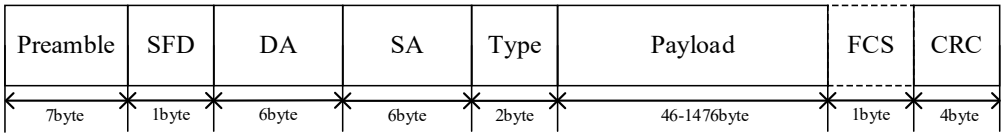

Figure 2: Time-triggered frame format diagram.

2.2.1. Encapsulate module design and implementation

The implementation of the time-triggered frame encapsulation module, that is, the receiving process of TT frame, the physical layer receives the incoming parameters and data from the PC and encapsulates them into the format of TT data frames. Create the packaging logic file TT_Pack_Frame.

The encapsulation module of the time-triggered frame is implemented by the encapsulation state machine, which defines the encapsulation state machine, including 6 states: Initial state (IDLE), data frame header state (Frame_Head), Ethernet header state (Eth_Head), udp header state (Udp_Head), received data state (Rec_Data), and single frame data transfer end state (Rec_End). The flag signal of the state jump is sj_flag. When sj_flag is 1, the state jump to the next state. Define false_en as a state jump error signal, and when false_en is 1, return the initial state (IDLE). Perform data beating tte_rx_data_reg[3:0] of the incoming module data tte_rx_data[3:0] on the PC side to improve the stability of data transmission, and use the data[7:0]data after concatenation as the actual input data. For the encapsulated module, the destination address is the address of the board, the source address is the address of the PC, and the service type of the time-triggered frame is 0x88d7. Define the mac_flag flag indicating that the destination MAC address is correct. If mac_flag is 1, des_mac and BOARD_MAC are consistent. If sj_flag is 1, the destination MAC address goes to the next state. When the input data enable signal tte_rxdv is at high power level, the TT frame encapsulation module starts to receive data frames. When tte_rxdv is at low power level, the single packet TT frame data is received and the state jumps to the initial state, waiting for the next time to trigger frame data to enable tte_rxdv signal.

After the TT frame encapsulation module is designed and implemented, the time trigger frame encapsulation module is verified by simulation. The TT frame package test file tb_TT_Pack calls the TT frame package module, stores the data that needs to be passed from the PC in txt file, and uses $readmemh to read into the data_mem register, data_mem is a width of 4, depth of 124 memory. Used to encapsulate the input of module data and parameters. Declares the input data start signal start_flag and enables it to be high. start_flag controls the enabling of tte_rxdv signals. When the data input enable signal tte_rxdv is valid, the 4bit data in data_mem is read to the input data tte_rx_data of PHY chip; Give the MAC address and IP address of the development board, run the simulation verification and check the simulation verification results.

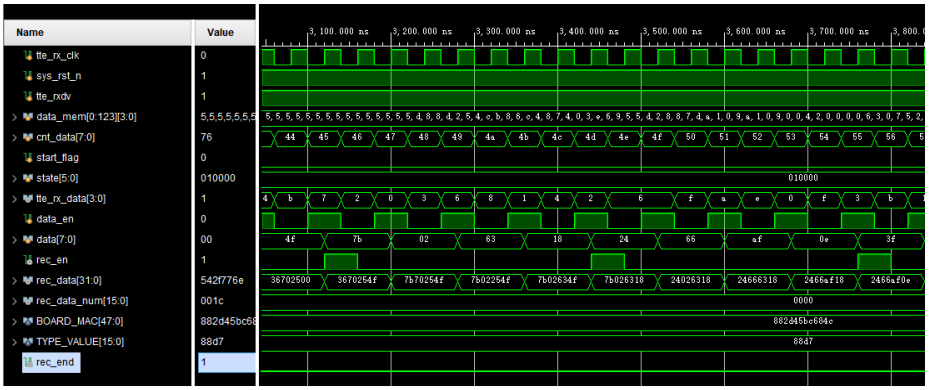

Figure 3: TT frame package module simulates waveform.

As shown in *Figure 3*, it is the simulation verification result of TT frame encapsulation test. From the verification result, it can be seen that the service type of frame is time-triggered frame 0x88d7. The data received by the physical layer is the data and parameters given from the memory data.txt at the PC end. The 4-bit input data tte_rx_data is converted into 32-bit output data rec_data and passed into the physical layer. That is, the memory data from the PC end is encapsulated into the format of time-triggered data frames to reach the received physical layer. The actual MAC address of the board is consistent with the input parameter, and the hopping between states of the encapsulation state machine is normal.

ISSN 2706-655X Vol.6, Issue 1: 125-130, DOI: 10.25236/IJFET.2024.060120

2.2.2. Design and implementation of the decapsulation module

Time triggers the implementation of frame decapsulation module, that is, TT frame sending process. The physical layer sends TT data frame to the PC and analyzes the parameters and data of the frame. Create the decapsulation logic file TT_UnPack_Frame.

The decapsulation module of the time-triggered frame is implemented by the parsing state machine, which defines the parsing state machine, including 7 states: Initial status (IDLE), ip header check status (Check_Sum), data header status (Packet_Head), Ethernet header status (Eth_Head), ip header and udp header status (Ip_Udp_Head), sending data status (Send_Data), and cyclic redundancy check Status (Crc). When send_en is set to 1, TT data frames are sent. Check_Sum status indicates the header check and check_sum calculation. Define the tte_tx_en signal. When the state is not the initial state and the ip header check state, the tte_tx_en signal is 1, indicating that the output data is valid. Ip_Udp_Head status sends a 20-byte ip header and an 8-byte udp header. The destination ip address is the ip address of the PC, and the source ip address is the ip address of the board. Send_Data is used to send valid data. The TT frame format requires that the minimum number of valid data bytes is 18 bytes. The supplementary byte counter is used to randomly supplement the missing bytes. When the PC receives the actual number of valid bytes, no operation is performed on the added bytes. Define the data read request signal read_data_req. When read_data_req is in high power mode, data from data_mem of the storage is read to send_data. Create a crc check file for the cyclic redundancy check module. Define send_end to send the end signal. When the status is Crc and cnt_send_bit is 7, send_end is 1, marking the completion of sending single packet data.

After the design and implementation of the TT frame decapsulation module, the time trigger frame decapsulation module is verified by simulation. The TT frame decapsulation test file tb_TT_UnPack calls the TT frame decapsulation module and CRC redundancy check module, gives the MAC address and IP address of the board and PC, and effectively sends data to be stored in the data_mem memory, which is a memory with a width of 32 bits and a depth of 3. When the read request signal read_data_req is valid, 4 bytes of data in data_mem is read to send_data to be sent. Set send_en to valid, run the simulation verification and check the simulation verification results.

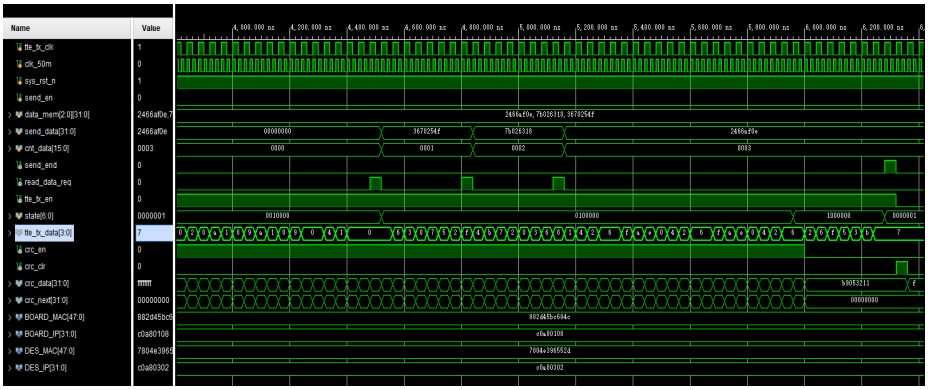

Figure 4: TT frame analysis module simulates waveform.

As shown in *Figure 4*, it is the simulation verification result of TT frame decapsulation test. In the decapsulated simulation waveform, the time-triggered data frame sent by the physical layer sends the parameters and data obtained after decapsulation to the PC. It can be seen that the data in the sent TT frame is consistent with the data obtained after decapsulation. The 32-bit data to be sent send_data is converted into 4-bit output data tte_tx_data and sent to the PC. That is, the format of time-triggered frames sent from the physical layer is decapsulated into 4-bit output data and sent to the PC.

2.3. Transmission control module design and implementation

Compared with GMII interface, RGMII interface is more simplified, with 4-bit data transmission interface and a high-speed transmission rate of 1000Mbps, it is one of the commonly used interfaces for Gigabit network communication. It is an interface between the MAC layer and the PHY layer and there is partial signal multiplexing. In order to carry out single-packet data caching, the FIFO module is created and invoked with the top-level module, and then the top-level module is verified by logic simulation. After the simulation and verification is completed, the PC is connected to the board through gigabit network cable and JTAG downloader, and the time-triggered transmission control logic is downloaded to the FPGA board through JTAG for board-level verification. The high-speed transmission of time

ISSN 2706-655X Vol.6, Issue 1: 125-130, DOI: 10.25236/IJFET.2024.060120

triggered network FPGA board is realized.

Create the simulation verification file tb_tte_rmii of the top-level module, call the top-level transmission control module, set the clock signal and reset signal, input the data from the txt file, and use $readmemh to read the parameters and data to the memory data_mem. When the input data enable signal tte_rxdv is valid, the 4-bit data in data_mem is passed into the PHY chip input data tte_rx_data. The bit width of tte_tx_data for sending data and tte_rx_data for receiving data is 4 bits, which conforms to the interface definition of RGMII. Run logic simulation verification, and the verification results are shown in *Figure 5* and *Figure 6*. From the simulation verification results, it can be seen that the TT frame data communication of the top-level module of the time-triggered transmission control based on the RGMII interface is normal, and the parameters of some TT frames received and sent are consistent with the data, and the module has passed the simulation verification.

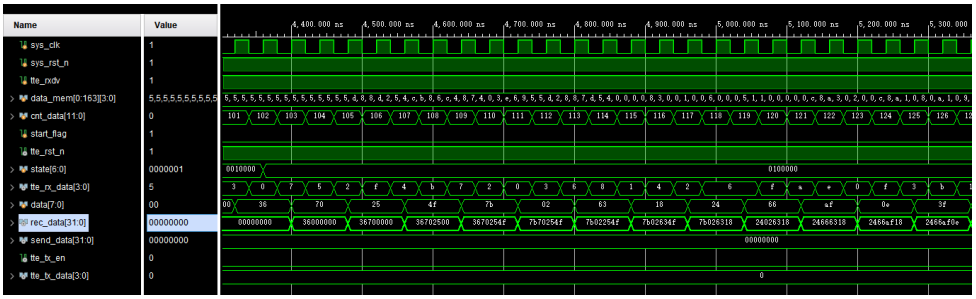

Figure 5: Time triggers the top-level network module to receive the simulation verification results.

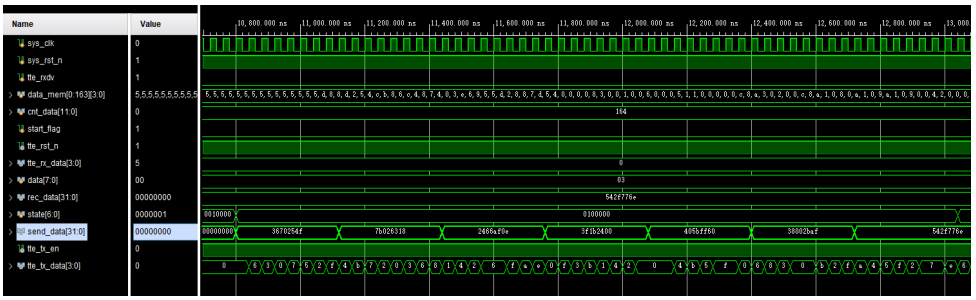

Figure 6: Time triggers the top-level network module to send simulation verification results.

After the module passes the logic simulation verification, the upper board verification is carried out, and the pins and pins are constrained and allocated according to the board pin diagram. After the pin constraint is complete, power up the development board and connect the PC to the board using the Gigabit network cable and JTAG downloader. The verification result on the upper board is shown in *Figure 7*. It can be seen that the destination IP address and port number are correct, the data on the sending end and the receiving end are consistent and correct, and the board level verification is passed.

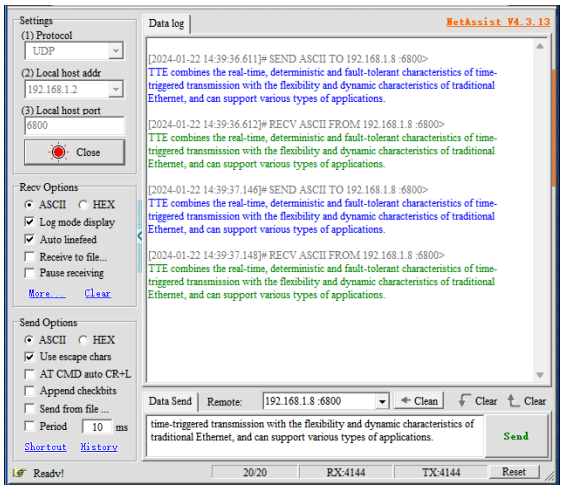

Figure 7: Board level verification results.

ISSN 2706-655X Vol.6, Issue 1: 125-130, DOI: 10.25236/IJFET.2024.060120

3. Conclusions

This paper studies the high-speed communication mechanism of time-triggered network based on RGMII interface, proposes the control design scheme of FPGA board for time-triggered top-level high-speed transmission, designs and implements the encapsulation module and decapsulation module for invoking the time-triggered frame of cyclic redundancy check module and configuration module, and the top-level module for time-triggered transmission control. Using RGMII interface transmission, the proposed time-triggered transmission control FPGA scheme is verified on board. Both the logic simulation and the board verification passed, and the data frame transmission rate in the time-triggered network based on the RGMII interface could reach gigabit, which realized the high-speed data transmission in the time-triggered network, and played a role in expanding the technical ideas for the subsequent research of time-triggered network.

References

[1] Gao P.F. Design of the Time-Triggered Ethernet Switch [D]. Xidian University, 2014.
[2] Zhou, J. Design and Implementation of AS6802 Synchronization Protocol Based on Time-triggered Ethernet [D]. Xidian University, 2018.
[3] H, Kopetz, G, Grunsteidl. TTP-A protocol for fault-tolerant real-time systems [J]. Computer, 1994, 27(1): 14-23.
[4] H, Kopetz, G, Bauer. The Time-Triggered Architecture [J]. IEEE, 2003, 91(4): 112-126.
[5] H, Kopetz, A, Ademaj, P, Grillinger. and K, Steinhammer. The Time-Triggered Ethernet (TTE) Design[C]// Eighth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing, 2005: 22-33.
[6] SAE AS6802. Time-Triggered Ethernet[S]. SAE Aerospace Standard, 2011.
[7] Lan, J., Zhu, X.F., Chen, Y. and Li, Q. Research on Time-Triggered Ethernet standards [J]. Aeronautic Standardization & Quality, 2013(05): 24-27+56.
[8] Zhang, K. Research on High Precision Clock Synchronization Technology of Time-Triggered Ethernet [D]. Beijing University of Technology. 2020.
[9] Lan, J., Xiong, H.G. and Li, Q. Clock synchronization fault-tolerance in Time-triggered Ethernet [J]. Computer Engineering and Design, 2015, 36(01): 11-16.
[10] Yang, J.X., Xu, Y.J., He, F. and Liu,Z.D. Research on Influence of Clock Synchronization Accuracy to TTE Real-Time Performance[J]. Electronics Optics & Control, 2016, 23(08): 33-38.
[11] Zhang, Y.J., Xiong, H.G., Liu, Z.D. and Li, Z. Application of TTE Communication Technology in Avionics System [J]. Electronics Optics & Control, 2015, 22(05): 49-53.
[12] Steiner, W. TTEthernet specification [J]. TTTech Computertechnik AG, Nov, 2008.
[13] Li, Z.W. The Research of Time Triggered Ethernet based on Network Calculus [D]. University of Electronic Science and Technology of China, 2013.
[14] Dong, J.W. Time Triggered Ethernet Accelerating Avionics Interconnection [J]. Electronics Optics & Control, 2016, 23(02): 74-78.