WEP Warwick Evans Publishing
International Journal of Computer Science and Information Technology
ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 2, Number 2, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v2n2.24
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Design and Implementation of Fc Data Forwarding

Changfeng Xiao[1, 2], Lin Li[1, 2, *]

[1]School of Electronic Engineering，Xi'an Shiyou University, Xi’an, Shaanxi 710065, China

[2]Key Laboratory of Oil and Gas Well Measurement and Control Technology in Shaanxi Province, Xi’an, Shaanxi 710065, China

*Corresponding Author: Lin Li

ABSTRACT

In order to meet the increasing data traffic demand of Fiber Channel, a FC data forwarding scheme based on Hardware acceleration technology is proposed. Firstly, an analysis was conducted on existing data forwarding schemes, and it was found that traditional software implementation methods often have performance bottlenecks. Therefore, it is proposed to directly process data forwarding using FPGA and use dedicated hardware modules to achieve high-speed processing of data frames to improve data forwarding speed. On this basis, a high-speed data forwarding scheme based on the FC protocol was designed and a system prototype was implemented. The FC simulation source was used to simulate the airborne environment for data forwarding experiments, which verified the effectiveness and feasibility of the design, significantly improving the speed of FC data forwarding.

KEYWORDS

Fiber Channel; Data Forwarding; Hardware Acceleration; Software Design; Simulation Verification

1. INTRODUCTION

FC (Fibre Channel) is a computer communication protocol specifically designed to meet the requirements for high-performance data transfer. It features high bandwidth, low latency, flexible topology, and distance-independence. Its transmission rate can reach up to 16 Gbps, providing excellent data transfer capability[1]. Due to its outstanding features, this protocol is widely used in the commercial sector and also supports avionics environments[2]. The high-speed nature of Fibre Channel is suitable for the demands of avionics environments. However, most onboard equipment currently does not support Fibre Channel. To address this, data acquisition recorders are used to convert Fibre Channel data frames into frame formats supported by various devices, thus achieving data transmission to the corresponding devices. As the demand for Fibre Channel increases, so does the volume of data being transmitted. Consequently, the data forwarding rate of traditional software can no longer meet the rapidly growing data requirements, and the software's forwarding rate directly affects the transmission rate of Fibre Channel.

In China, research on Fibre Channel is primarily conducted by universities and research institutes, mainly applied in Fibre Channel Storage Area Networks (SANs), achieving partial functionality of Fibre Channel networks and yielding some theoretical and simulation results. Zhang Min has researched and designed the data interface of the data forwarding layer for FC switch line cards[3]. Zhu Zhiqiang, combining Fibre Channel-related protocol standards, proposed a communication system architecture based on Fibre Channel to address the long-distance transmission needs of DVI signals in complex avionics environments, providing a solution for long-distance transmission of DVI signals via Fibre Channel[4].

This article improves the FC data forwarding function. The traditional method of data forwarding involves the CPU issuing commands for hardware to read the data frames to be forwarded from registers and then send them. This process requires calling a system interrupt for each data frame, significantly reducing forwarding efficiency. Therefore, a hardware acceleration method for data forwarding is proposed. Once data is stored in registers, it no longer requires CPU instructions for forwarding; instead, the hardware performs the forwarding automatically. Verification has shown a significant increase in the rate of data forwarding, pointing to a new direction for enhancing FC forwarding rates.

2. ANALYSIS OF FC AND RELATED PROTOCOLS

2.1. Overview of FC Protocols

The FC protocol is a high-speed serial data transfer protocol based on a serial channel architecture, with transfer rates up to 16Gbps or even higher. It is used in high-performance computing and large-scale storage domains, extensively applied in SAN (Storage Area Networks) and other scenarios requiring high-speed data transfers[5].

The FC protocol offers a fast, scalable, and reliable means of transmission, supporting various types of traffic and topology structures. It has several advantages: First, it supports multiple topologies such as point-to-point, loop, and fabric. Second, it supports various concurrent traffics, including SCSI, conventional data, and infrared transmission. Third, it enables efficient bandwidth utilization, increasing data transfer efficiency. Fourth, it provides hardware-based management and security controls to ensure the safety and integrity of data transfers[6].

2.2. The layered structure of the FC protocol

The layered structure of the FC protocol includes the physical layer, transport protocol layer, network layer, application layer, and protocol mapping layer, each with corresponding control and management mechanisms, as shown in Figure 1. At the physical layer, the FC protocol uses optical fiber as the transmission medium, connecting physically through fiber switches. In the transport protocol layer, the FC protocol employs a frame structure similar to Ethernet for data transmission, ensuring the reliability and smoothness of data transmission through frame identification and traffic control mechanisms[7]. At the network layer, the FC protocol provides routing and broadcasting functions to meet the needs of complex data transmissions. In the application layer, the FC protocol mainly offers generic services for multi-port nodes, including basic link services and extended link services. In the protocol mapping layer, high-level protocols related to channels and networks such as SCSI, IP, HiPPI, ATM are defined for mapping to lower layer protocols, akin to the transport layer in the OSI model.

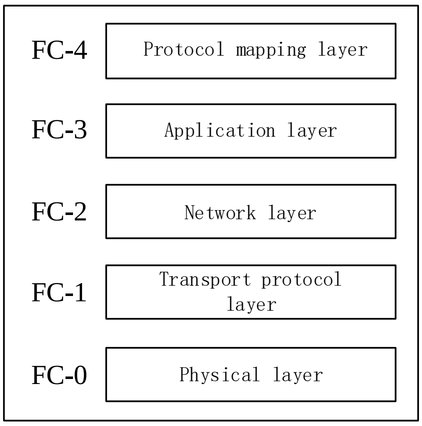

Figure 1 The structure of the FC protocol

2.3. FC frame format

In FC network transmission, the basic unit of data is the FC frame, whose functionality is defined in the FC-2 layer. A complete FC frame format, as shown in Figure 2, comprises four parts: the Start Of Frame (SOF), Frame Header (Head), Payload, and End Of Frame (EOF), where the payload stores the data and related information to be transmitted.

SOF	Head	Payload	EOF

Figure 2 FC frame format

1) SOF: This is 4 bytes long, representing the start of a frame. When these 4 bytes are recognized, it indicates the beginning of frame reading.
2) Frame Header: The frame header is 24 bytes long, with specific content as shown in Figure 3. R_CTL (Route Control) is used to categorize the frame's function. D_ID and S_ID represent the destination and source addresses, indicating the origin and destination of the FC frame. CS_CTL/Priority is used for specific class control, generally for service types and priority handling, but can be ignored in general FC frames. TYPE identifies the specific type of frame, typically used in conjunction with R_CTL to assist in FC addressing and routing. F_CTL represents the frame control field, containing control information about the frame, including frame and sequence position information, padding bits, etc. SEQ_ID is the sequence ID, used to identify a unique FC sequence. DF_CTL is the data field control, describing optional headers in the payload. SEQ_CNT is the frame count, distinguishing different frames within the same sequence and indicating whether any frames have been lost in reception. OX_ID and RX_ID are identifiers for the sending and receiving exchange ports, both unique and predefined. Parameter is the parameter section in the frame header, generally used to record data offset information.
3) Payload: The payload length ranges from 0 to 2112 bytes and contains the main content of the frame, serving as the storage location for user data. Typically, there are 4 bytes of CRC (Cyclic Redundancy Check) frame check sequence at the end of the payload, used to verify the accuracy of the frame contents.
4) EOF：End Of Frame indicator, 4 bytes in length, signifies the end of a frame.

Bits / Words	31...24	23...16	15...8	7...0
0	R_CTL	D_ID		
1	CS_CTL/Priority	S_ID		
2	TYPE	F_CTL		
3	SEQ_ID	DF_CTL	SEQ_CNT	
4	OX_ID		RX_ID	
5	Parameter			

Figure 3 Format of the FC frame header

2.4. Ethernet and TCP/IP

Ethernet technology is a type of Local Area Network (LAN) technology, one of the most widely used LAN technologies today. It enables data transmission and communication between computers, servers, switches, and other network devices. The IEEE 802.3 standard establishes the protocols for Ethernet technology, including specifications for physical layer connections, electrical signals, and media access control protocols. Ethernet has largely replaced other LAN standards, such as token ring, FDDI, and ARCNET, and is the most commonly applied LAN technology.

The TCP/IP protocol stands for Transmission Control Protocol/Internet Protocol, a generic network communication protocol. It encompasses a suite of protocols including FTP, UDP, TCP, SMTP, IP, etc., among which the TCP and IP protocols are the most representative. The TCP/IP protocol structure is inspired by the OSI model and is divided into four layers from top to bottom: the application layer, transport layer, network layer, and link layer. This structure merges the OSI model's data link layer with the physical layer into a single link layer and omits the presentation and session layers[8]. The TCP/IP protocol is one of the most commonly used and crucial core protocols in internet data transmission, widely applied across businesses, educational institutions, government agencies, and more.

2.5. UDP protocol

The UDP protocol, also known as User Datagram Protocol, is a connectionless, unreliable data transmission protocol. It utilizes IP protocol datagrams for data transmission without establishing a connection, offering no guarantees for reliable data transfer or mechanisms for retransmission. The UDP protocol is mainly used in applications that require fast data transmission where the reliability of data transfer is not a high priority. Positioned within the transport layer of the TCP/IP protocol stack, it lacks the message acknowledgment mechanisms, flow control, and other features found in the TCP protocol. By sacrificing data reliability, UDP achieves extremely high transmission efficiency, making it indispensable in environments where real-time requirements are exceptionally stringent.

Upon receiving a message from the application layer, UDP encapsulates the data into UDP packets, adding only an 8-byte UDP header, as opposed to the 20-byte TCP header, thus saving significant space and enhancing transmission efficiency[9]. After packet formation, data is directly transmitted to the network layer without the need to establish, acknowledge, or release connections. This connectionless transmission method significantly reduces waiting times in data delivery. Figure 4 illustrates the UDP forwarding process[10].

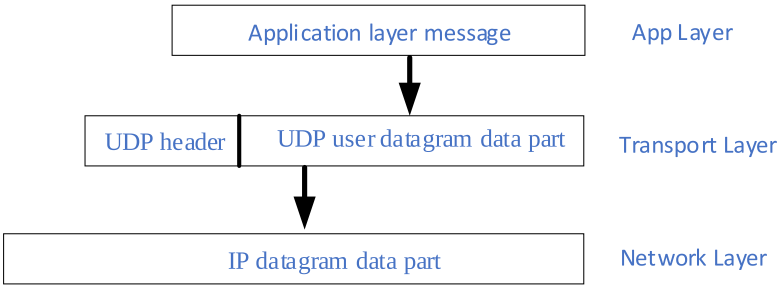

Figure 4 UDP forwarding process

3. FC FORWARDING FUNCTION DESIGN

With the continuous progress of society and the development of science and technology, various transmission protocols have emerged in network technology. The different characteristics of each protocol allow them to play their roles in various fields. For example, the FC (Fibre Channel) protocol is widely used in aerospace and aviation environments due to its high-speed transmission and high reliability. However, the FC protocol relies on optical fiber transmission, which, due to its higher cost and the fragility of fiber optic cables, results in installation challenges and a loss of flexibility. On the other hand, the advantage of the UDP (User Datagram Protocol) lies in its speed and flexibility, sacrificing stability and security for high-speed data transmission. Since it uses copper cables for transmission, connections can be easily established anywhere. Therefore, it is necessary to combine the two protocols, utilizing the conversion between them to enhance the stability and flexibility of network transmission, enabling their application in various scenarios[10].

3.1. Overall design

The FC protocol is generally used in avionics systems. Since the onboard power supply is uniformly 28V DC, the voltage for each module is standardized to 5V. Power conversion from 28V to 5V is managed by a power module to supply each module, as illustrated in Figure 5. To ensure the provision of reliable flight parameters, a redundancy method for data transmission is employed, meaning data is sent over two separate optical fiber channels. The two streams of data are stored and forwarded separately in data reception modules, significantly enhancing the reliability of data forwarding and reducing the forwarding of erroneous frames.

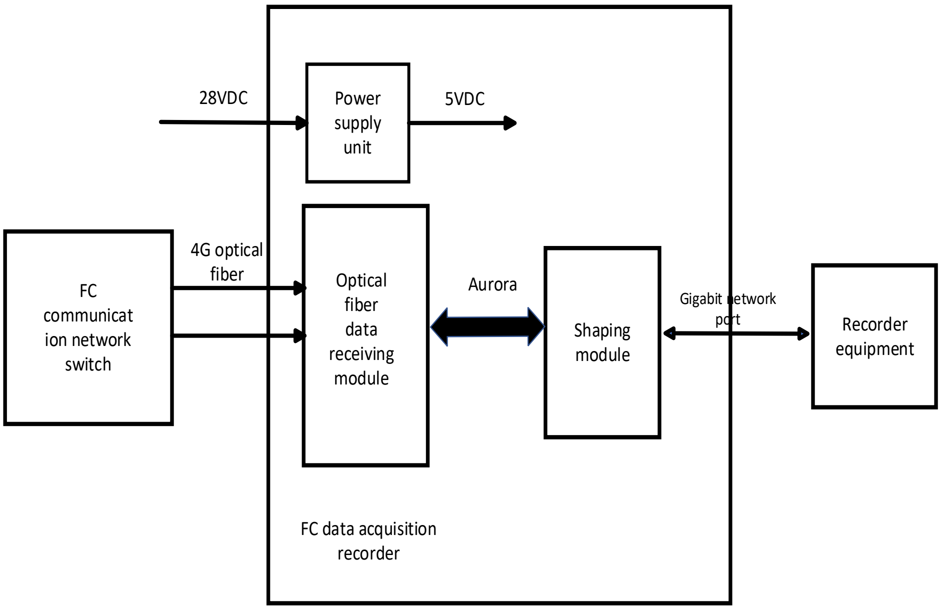

Figure 5 FC data acquisition recorder system block diagram

FC data is first transmitted to the fiber data reception module via the optical path, then transferred to the shaping module through the Aurora and PCIe interfaces. The shaping module consists of a CPU and a shaping board, with the CPU board handling the caching, packaging of characteristic parameters, and Ethernet transmission functions. The fiber data reception module filters the received FC frames and sends them to the CPU board's memory via the PCIe interface. Application software reads the data from memory for characteristic parameter extraction and data packaging processing, and after packaging, the data is sent to the shaping board via an Ethernet interface. The shaping board receives the data packets sent by the CPU board, performs data caching, and sends them to external devices over Ethernet at configured intervals. Additionally, the shaping board receives FC frame data sent by two data reception modules through two Aurora interfaces, performs frame slicing and packaging, and achieves 100% forwarding via Gigabit Ethernet interface in UDP mode.

3.2. FC Forwarding Module Design

The implementation of traditional FC forwarding functions mainly relies on software for converting FC frames to UDP frames. However, as the demand for data volume increases, relying solely on software to process frames significantly reduces the efficiency of FC forwarding. Therefore, a solution that combines hardware acceleration with software processing has been adopted to improve FC forwarding efficiency. As illustrated in Figure 6, once the shaping module receives an FC frame, it first sends it to the PL side (Programmable Logic), which is the FPGA core. The PL side checks and verifies the received FC data frame's protocol type and data length, filtering out erroneous frames in the process. After verification, the FC frame undergoes protocol conversion and frame splitting. The processed UDP frames are then sent to the PS side (Process System) via DMA, stored in DDR, and finally, software processes the UDP frames stored in DDR and forwards them to recording devices via Ethernet to complete FC data forwarding.

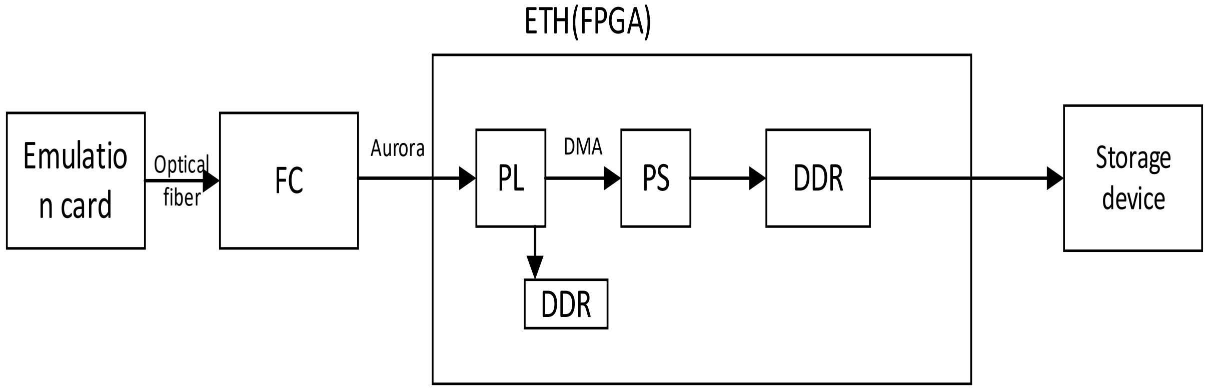

Figure 6 FC forwarding function

FC data is stored in external DDR via the AXI interface. Under the principle of fair scheduling and when the DDR cache is not empty, the two received FC frames are read from DDR and written into the data reception buffer, with channel identifiers marked upon writing into the data reception buffer. The reception buffer employs a double-buffer structure, each of 256KB size. When one buffer is full, it is transferred to the PL side via DMA for frame format conversion and frame slicing. The converted data, sized at 0x800 (2K) per frame, is sent to the PS side, packaged into UDP protocol, and stored in a 512MB DDR for forwarding in a circular queue manner.

Since the length of FC frames ranges from 36 to 2148 bytes, and UDP frames from 46 to 1500 bytes, FC frames exceeding a certain length require frame slicing to meet the length requirements of UDP frames. Therefore, FC frames longer than 1400 bytes are split, with the first 1400 bytes as the first UDP frame and the data following 1400 bytes as the second UDP frame. To distinguish whether data frames are split, a frame splitting identifier field is added to the output. Figure 7 shows the data format after forwarding, using frame counting to record the number of frames sent, a frame splitting identifier to determine whether the frame is split and whether the data is the first or second frame, and checking the transmission status to confirm if the frame has errors, with the rest being the data's inherent information.

Frame count 2Byte	Frame identification word 2Byte	IDentified word 4Byte	Channel number 1Byte	Reserve 1Byte	FC frame count 2Byte	Data length 2Byte	Transmission state 2Byte	System time scale 8Byte	Local time scale 8Byte	Date scale 8Byte	FC frame 36~2148Byte

Figure 7 FC data forwarding output data format

For redundancy handling of two-way FC data, a "first come, first served" rule is applied. When the receiver receives the first frame of a certain message ID, the frame count of that channel frame is recorded in the frame count register. Upon receiving data of the same message ID, the new frame count is compared with the value in the frame count register. If the new frame count is less than or equal to the value in the register, it is considered as already received and redundantly discarded. If the frame count is greater than the value in the register, the frame is accepted and the register value is updated.

4. SIMULATION AND VERIFICATION TEST

4.1. Test Platform

The test platform consists of a PC, an FC simulation card, and a hardware development platform. The FC simulation card, inserted into the PC motherboard, simulates the transmission of FC data from a transmitter. The simulation card is connected to the hardware development platform via an optical fiber, sending FC data to the device. After the device performs protocol conversion, UDP data packets are forwarded to the PC via an Ethernet interface for storage. Subsequently, Wireshark software is used on the PC to examine the contents of the forwarded data.

4.2. FC Forwarding Function Test

After setting up the FC data forwarding test platform, it is necessary to test the functionality and performance of the FC data forwarding, including the verification of long-frame splitting, data forwarding integrity, and forwarding rate under extreme conditions. Since it's challenging to fully simulate the onboard environment in a laboratory, an FC simulation card is used to simulate the onboard environment for sending FC data. The FC simulation card allows for setting the content, length, Message ID, sending cycle, number of sends, and number of sending messages on the PC. Initially, data is configured to start with the message ID 00000100, setting up 40 incremental data entries with null values, a sending cycle of 10ms, and testing at a node rate of 4.25Gbps.

After sending data for thirty minutes, Wireshark reveals no data types other than UDP, and data of all 40 message ID types can be received, as shown in Figure 8. The SOF and EOF segments of the data are examined, with SOF as BC B5 56 56, and EOF as BC 95 75 75, matching the set values, thus verifying the integrity of the data frame. Since the forty data entries are consistent in length and are long frames, the received data is also consistent in length and includes frame splitting. The continuous increase in frame counting proves that no frames were lost.

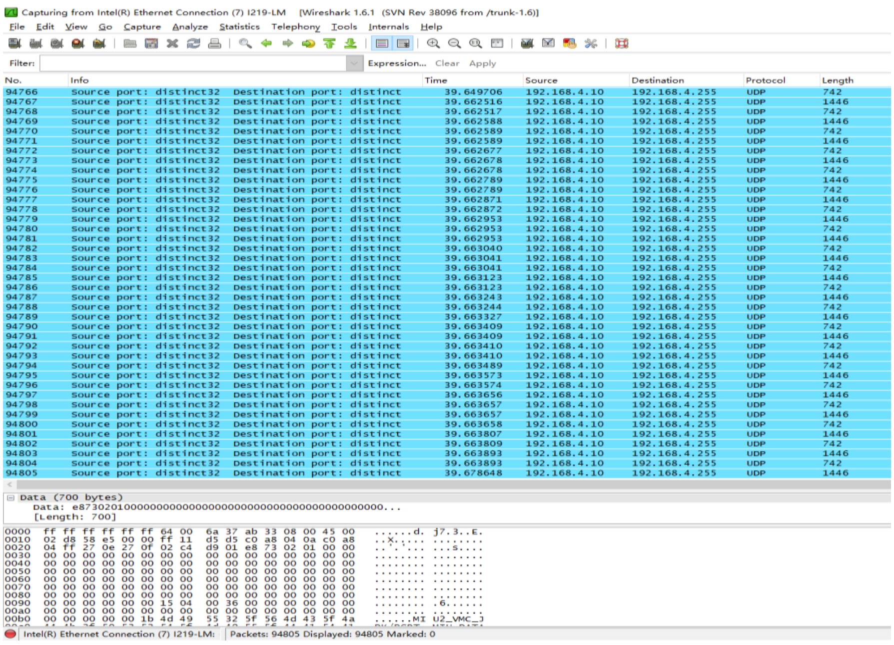

Figure 8 Thirty minute data forwarding test

Further verification of the frame splitting functionality is performed by sending long and short frames and verifying the frame length and frame splitting identifier through Wireshark. The length of the first frame of a long frame is 1446 bytes, as it is set to split for lengths greater than 1400 bytes, and with a UDP padding field of 46 bytes, the total 1400+46=1446 meets the splitting length requirement. The frame splitting identifiers of the first and second frames of a long frame also comply with the regulations. Figure 9 shows the UDP packet of a short frame, with its frame splitting identifier also meeting the standards, confirming through experimental testing that the frame splitting functionality is normal.

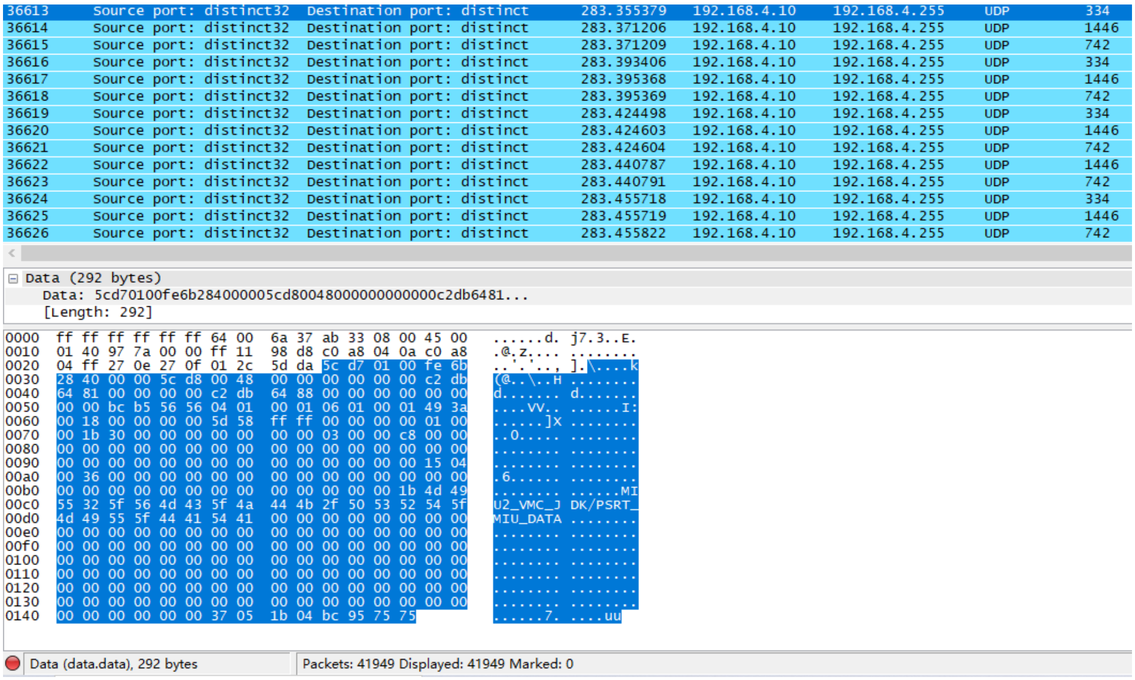

Figure 9 Short frame data

Finally, the FC forwarding rate is tested by sending 256 messages. The software measures the FC forwarding rate to be approximately 37.32MB/s, as shown in Figure 10. Compared to the pre-improvement forwarding rate of about 15MB/s, this represents an increase of approximately 2.5 times, significantly validating the success of the hardware acceleration scheme.

联合ID	源ID	目的ID	主题ID	消息ID	消息名称	消息类型	发送次数
1	1	1	1	1	1	周期	174
2	2	2	2	2	2	周期	174
3	3	3	3	3	3	周期	174
4	4	4	4	4	4	周期	174
5	5	5	5	5	5	周期	174
6	6	6	6	6	6	周期	174
7	7	7	7	7	7	周期	174
8	8	8	8	8	8	周期	174
9	9	9	9	9	9	周期	174
10	10	10	10	10	10	周期	174
11	11	11	11	11	11	周期	174
12	12	12	12	12	12	周期	174
13	13	13	13	13	13	周期	174
14	14	14	14	14	14	周期	174
15	15	15	15	15	15	周期	174
16	16	16	16	16	16	周期	174
17	17	17	17	17	17	周期	174
18	18	18	18	18	18	周期	174
19	19	19	19	19	19	周期	174
20	20	20	20	20	20	周期	174
21	21	21	21	21	21	周期	174
22	22	22	22	22	22	周期	174
23	23	23	23	23	23	周期	174
24	24	24	24	24	24	周期	174

Figure 10 FC forwarding rate

4.3. Summary of this chapter

This chapter verifies and tests the FC data forwarding functionality and performance. According to the test results, the design objectives for FC data forwarding have been achieved, and there has been a noticeable improvement in forwarding rate. Overall, the design scheme has essentially met its design goals, but there are still many areas that require further improvement and refinement.

5. CLOSING REMARKS

This study designed and improved the FC forwarding function, implementing FC data forwarding with hardware acceleration to enhance the software's frame processing capability. Simulation experiments have verified that this can significantly increase the FC data forwarding rate. However, the forwarding efficiency still has room for improvement, which could be further enhanced by methods such as reducing the content of the data frame header. With the increasing demand for FC data transmission, future requirements may include the transmission of video and audio data. This research still has potential for improvement, such as through collaborative processing of forwarding frames by software and hardware to enhance processing efficiency, which will be practiced in future studies.

REFERENCE

[1] Guo Meng, Zhang Hongbo, Liu Zhu, Xue Ning, Liang Shuo. Research and Application of a New Generation of High Speed Optical Fiber Bus Based on Rocket [J]. Computer and Digital Engineering, 2022,50 (07): 1545-1551.

[2] Tian Ze, Xu Wenlong, Xu Heng, Li Xiaoyu. The research of fiber channel technology [J]. Electronic Technology Applications, 2016,42 (09): 143-146.

[3] Zhang Min, Qian Liang, Zhao Shikun, et al. Research of interfaces in line-card data transmission of FC switch [C]. Proceedings of the 6th National Conference on Signal and Intelligent Information Processing and Application, 2012:306-309+351.

[4] Zhu Zhiqiang, Wang Shikui, Sun Dongxu, Wu Jian. Research on DVI Signal Transmission Technology Based on Fibre Channel in Airborne Environment [J]. Aviation Computing Technology, 2022,52 (01): 118-121.

[5] Xu Yujie, Meng Bo, Zhang Nan. Design and Implementation of an FC Protocol Testing Equipment [J]. Computer Knowledge and Technology, 2021,17 (16): 43-45.

[6] Wang Fei, Huang Panpan. FPGA-based FC and Gigabit Ethernet bridge design [J]. Radar and Countermeasures, 2022,42 (01): 58-64.

[7] Wang Bo. Research on FC-UDP/RocketIO protocol processing and conversion technology [D]. Zhejiang University, 2017.

[8] Yang Xiaofan. TCP/IP related protocols and their applications [J]. Communication World, 2019,26 (01): 27-28.

[9] Li Huimin, Yan Minghui, Ren Hongbin, Fu Shixun. Secure and Efficient Data Transmission Based on UDP Protocol [J]. Journal of North China Aerospace Institute of technology, 2022,32 (06): 8-10.

[10] Li Dongyu, Design of UDP Communication with Flow Control [D]. Xidian University, 2021. DOI: 10.27389/d.cnki.gxadu.2021.003320.

[11] Chang Yuan, FC/UDP High Perfermance Hardware Acceleration of Data Link and Research on Technology of Dual-Redundancy [D]. Zhejiang University, 2022. DOI: 10.27461/d.cnki.gzjdx.2022.002023.