CS120 Project Report

Liu Yifei   Li Shuhang
School of Information Science and Technology, ShanghaiTech University

I. Introduction

In the four Projects of CS120, we borrowed the design thinking of computer networks, based on audio hardware, and used the operating system interface layer to implement an information transmission network using sound as the information transmission medium.We call it Athernet. It is worth noting that the entire network architecture is designed to be broadly consistent with the OSI model. We designed and built a complete network system running at the physical, data link, IP, transport and application layers. Most notably, it interconnects well with the Internet, and by implementing the ICMP and FTP protocols in Athernet, our network can be seamlessly embedded in the Internet to perform specific tasks.

II. Project 1: Acoustic Connection

In this project, we need to implement the modulation and demodulation of binary information from the computer to the sound waves in the air. Overall, the two computers running the system are able to transmit data through sound. Specifically, the sender encodes the information into an audio signal and sends it out through the computer's speakers, and the receiver receives the audio signal through a microphone and demodulates the sent information from it. Fortunately, the technology of modern radio is very advanced, and we can refer to the experience of centuries of engineering practice. At the same time, radio transmission is still very popular in our time, so we can draw on and use the latest technology to reduce the difficulty of development. Unfortunately, the project required a high performance and accurate sound transmission technology in the human ear's sound band. The main difficulty we encountered was that the normal audio band is more vulnerable to interference in real life. The different environments in which sound is transmitted lead to different acoustic field models, and the way sound is transmitted is uncertain and unstable. Generally speaking, modern operating systems are not designed for high performance sound communication and therefore provide only general sound playback and recording capabilities. Compared to specialized equipment designed to process audio signals, we need to bear the data variations of audio signals parsed by audio hardware after passing through the operating system on our own.

At the technical implementation level, a low latency and high performance audio middleware is crucial, and in practice we chose the ASIO library, which offers capabilities that provide overall feasibility in theory. Regarding the modulation mode, we chose the simple and easy FSK modulation, where we specified that bit 0 would be modulated as a 5 ms 8000 hz sine wave and bit 1 would be modulated as a 12000 hz sine wave, taking into account the continuity of the audio playback, an additional continuous matching was done to generate a sine wave for a piece of data. The header of each frame is a sine wave superimposed with increasing and then decreasing frequency. The length of a data frame is limited to 100 ms. At the receiver side, the header is continuously used for convolutional matching and when the header is detected, the audio of the next frame length is saved for discrete fast Fourier transform analysis. We finally succeeded in transmitting 10,000 bits in 10 seconds.

III. Project 2: Multiple Access

In project 2, the biggest difference is moving from wireless link to wired link. The advantage of this is to reduce the heterogeneity and complexity in acoustic hardware. This also makes the transmission more stable, greatly avoiding the interference of noise and echo. In order to ensure the transmission rate, we use 5 sample points to represent one bit, and enrich the structure of the packet. The following fields were addedthe 10-bit seq field, the 2-bit frame type field(distinguish data frames from ACK frames), the 2-bit MACaddress field, and the 16-bit CRC check field (with the CRC16 check) to achieve more complex transmission requirements.

For each Athernet node, we use multithreading to ensure the simultaneous transmission and reception (half duplex) of two nodes, and use the state machine to implement the transfer and reception state switch. We also support the use of MacPing and MacPerf commands to test Athernet's delay and bandwidth. And has made

active and effective attempt to CSMA protocol.
In the specific implementation, we initially chose 20 bits of 1 in a row as the packet header, and determined the position of the packet header by detecting the energy of 20 consecutive bits after a bit 1. We also tried to intercept the middle three sample points and unpack by taking the maximum common multiple of 5 of the number of consecutive high and low levels, and achieved good results before part2. However, after adding the influence of noise, it is difficult to judge the position of the header in the form of measuring header energy. Inspired by lecture, we try to use the form of alternating 0/1bit as the header, and adjust the length of payload, so as to ensure the integrity of the data segment as much as possible. Although the check of part3 didn't succeed in the end due to many factors such as threshold value or noise volume, in our own debugging, we have been successful for many times and can detect noise relatively stably.This is also something we should reflect on. In the last two parts, we complete the detection of throughput and RTT by implementing two special data frames.

IV. Project 3. Gateway

In project 3,we build a gateway for the Athernet. In this way, Athernet devices are able to connect to the Internet.Thus,the interconnection between different protocol nodes is achieved.The intermediate node2 acts as NAT to reads the ip address and data contained in the frame, and then repackage the UDP frames to forward to the corresponding IP address node. This is also our core implementation. We also adopt some special methods such as using French characters as the end of a message, node1 sending refined ICMP packets and node2 repackaging to send standard ICMP packets, which are more subtle implementation of the project's checkpoint. It helped us gain a deeper understanding of socket programming.

V. Project 4. FTP

In project 4,we implement an FTP client on Athernet. Our client is able to support basic commands.First, the input commands are identified and corrected by finding the maximum common subsequence to ensure the authenticity and validity of the commands. The commands are transmitted in the form of data packets. After receiving the commands, the ftp server performs corresponding operations.For commands that return output, the receiving thread of node1 can capture the frames and display the contents. We also adopted the method of termination frame to monitor the end of data transmission. The termination frame is a special data frame. We optimized its structure and adopted better data filling to avoid attenuation caused by excessive continuous same level and ensure the correctness of transmission without ACK frames, which allows node1 to correctly recognize the end of a file or command being sent.

VI. Advice and Suggestions

First of all, we have always believed that Project 1 has too high requirements on hardware. We spent a lot of time on parameter adjustment and changing modulation modes, but finally, we did not achieve an ideal result although we think the code implementation is sufficient to meet the requirements of the project.Different hardware and different modulation modes may lead to the unfairness of the project to some extent.
As for the score setting of different parts, it is relatively reasonable on the whole. Some difficult questions are selected with low score and as optional parts, which not only guarantees extra benefits for students who are willing to challenge, but also prevents other students from having score panic. However, the whole optional part with high score and the total score without upper limit may lead to a large difference in the scores of different groups. In our opinion, it is a better choice to set a limit on the total score of the project or reduce the overall score of the optional part and reduce its overall difficulty.This may stop students from malicious competition and let bonus part plays true role in this project.

VII. Every End is a New Beginning

In these four projects, what impressed us most is the word trade off. For example, how to ensure the transmission rate on the premise of ensuring the correctness, and how to choose the more suitable solution among the different solutions that are simple or more beneficial to the subsequent parts,which require us to think about along the process. The choice of modulation mode in project1, the design of competition mode in project2, the implementation of NAT in project3, etc.All require us to have high foresight and integrity in the macro layout of the entire network implementation, which will provide exercise for our future engineering-oriented programming practice.