

# Athernet: Acoustic-based User Space Network Stack

Cheng Peng

ShanghaiTech University

pengcheng2@shanghaitech.edu.cn

Huizhe Su

ShanghaiTech University

suhzh@shanghaitech.edu.cn

## Abstract

Athernet is a user-space TCP/IP stack with full functionality and handy utilities built on the acoustic channel. This report describes the main architecture design and the technical implementation of the Athernet.

**Keywords:** network stack, acoustic link, network protocol

## 1 Introduction

This work is submitted in partial fulfilment of the course CS120: Computer network at ShanghaiTech University. The network stack is an essential topic in computer networks and has various implementations. To comprehensively understand it, we are asked to implement our network stack Athernet.

Unlike traditional wireless or wired methods, Athernet uses sound waves to transmit data between devices. The stack has implemented the basic functionalities of the standard TCP/IP and can be used with existing network infrastructure. Additionally, the system has the potential to be used in areas where wireless communication is not feasible, such as underwater communications.

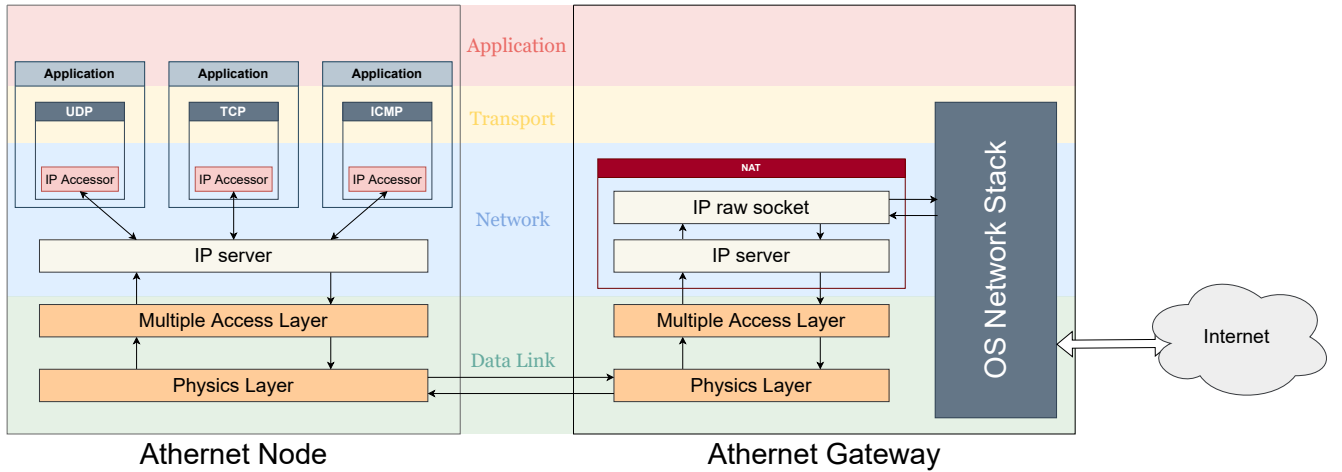


Figure 1: the overview architecture of the Athernet network stack

The network stack is composed of the following four layers where each layer is built on the previous one.

- **The link layer:** The link layer provides reliable bidirectional data transmission in local area. It is decomposed into two sublayers namely the *physics sublayer* and the *media access control sublayer*. The physical layer is responsible for the converting digital data into audio signal or the reverse process and sending/receiving data through sound waves. It supports both noisy wireless medium and audio cable media. Two different modulation schemes are implemented for the two scenarios: a passband modulation scheme that combines *BPSK* and *OFDM* for the wireless scenarios and a baseband modulation scheme using *4B5B* with *NRZI* for the cable connected scenarios. The media access layer deals with collisions that may happens when multiple nodes sends signal to the media simultaneously. *CSMA/CA* is implemented to provide reliable data transmission.
- **The network layer:** *Internet Protocol(IP)* is implemented for inter-network data transmission. An daemon process called the *IP server* regularly sends packets to peers and receives packets from peers. To allow programs in the next layer to access

the network layer service, *IP accessor* which communicates with the *IP server* using unix domain socket is created. The *network address translation(NAT)* is also implemented in this layer, which enables the communication between Athernat LAN and the Internet.

- **The transport layer:** Various transport layer protocols are implemented including *transmission control protocol(TCP)*, *user datagram protocol(UDP)* and *internet control message protocol(ICMP)*. Athernat socket, a set of APIs that resembles the socket API in rust standard library, are exposed for applications to use the transport layer services.
- **The application layer:** A simple *file transfer protocol(FTP)* client supporting a set of restricted control commands and only passive transmission mode is implemented on Athernat TCP to demonstrate the capability of Athernat. For Athernat to support the pre-existing applications which are built on OS TCP/IP stack, a SOCKS5 proxy server is implemented that redirects network traffics through Athernat.

Figure 1 shows the overall architecture of the Athernat. The Athernat nodes are interconnected and can connect to the Internet through the Athernat gateway, which runs a NAT.

The paper is divided into five parts. Through chapter two to five, we describe the design of the four layers of our network stack. In the last chapter, we analyze the problems in our design and make a technical summary.

## 2 Data Link

The foundation of the entire network stack is the data link layer which provides reliable bidirectional packet transmission between two nodes in a shared noisy acoustic medium. The data link layer consists of the physics sublayer and the medium access control sublayer.

### 2.1 Physics Sublayer (PHY)

Built on the audio I/O library, PHY enables basic data transmission with no delivery or integrity guarantee. A PHY frame is a sequence of PCM audio signal samples that represents a chunk of bytes. It is the transmission unit on this layer. Each frame contains a fixed preamble signal at the beginning for detection as well as synchronization and a payload signal that encodes 0/1 bits.

For data transmission, PHY layer constructs a frame from a chunk of bytes and send it through the medium. For data receiving, PHY layer repeatedly pulls audio samples from the medium and find the beginning of PHY frames. When a frame is identified, the payload signal is extracted and decoded.

#### 2.1.1 Preamble Signal Design

A chirp is a signal whose instantaneous frequency increases or decreases with time.  $x_p(t)$  is a linear chirp signal whose lowest and highest instantaneous frequency are  $f_a$  and  $f_b$  respectively.

$$x_p(t) = \begin{cases} \pi \frac{f_b - f_a}{T} t^2 + 2\pi f_a t & t \in [0, T] \\ \pi \frac{f_a - f_b}{T} (t - T)^2 + 2\pi f_b (t - T) & t \in [T, 2T] \end{cases}$$

Chirp signal has a distinct frequency-domain feature so it won't interfere background noise and the carrier wave. Auto-correlation energy peak detection is used to find the starting position of the preamble signal which is followed by the payload signal.

#### 2.1.2 Modulation and Demodulation

In order to transmit and receive digital signals, a modulation scheme has to be implemented. Air and audio channels are media of different characteristics, so they need different modulation schemes.

For wireless scenario, background noise is strong from 0Hz to 6000Hz, so a passband modulation scheme combining binary phase shift keying (BPSK) and orthogonal frequency-division multiplexing (OFDM) is used. BPSK encodes bits with phase changes:

$$\begin{aligned} x_0(t) &= \sin(2\pi f t) \\ x_1(t) &= \sin(2\pi f t + \pi) = -\sin(2\pi f t) \end{aligned}$$

are used to represent 0 bit and one 1 respectively. To recover a bit from audio signal, consider the dot products  $\langle x_0(t), x_0(t) \rangle$

and  $\langle x_0(t), x_1(t) \rangle$ :

$$\int_a^b (x_0(t) \cdot x_0(t)) dt = \int_a^b x_0^2(t) dt > 0$$
$$\int_a^b (x_0(t) \cdot x_1(t)) dt = - \int_a^b x_0^2(t) dt < 0$$

A zero bit is produced if the dot of the received samples and samples of  $x_0(t)$  is positive, otherwise a one bit is produced.

To increase the utilization rate of the spectrum, OFDM of  $N = 64$  is introduced. However, unlike radio medium, transmitting real part and imaginary part of a complex number simultaneously is not possible, so only half of the frequency channels can be used to encode bits. Also, several channels of extremely low and high frequency are not usable as background noise is too strong.

As for cable-connected scenario, baseband transmission is feasible as low frequency background noise no longer exists. 4B5B + NRZI enables higher bandwidth and lower error rate.

## 2.2 Medium Access Control sublayer

The medium access control layer regulates access to the shared medium and implements error detection and retransmission mechanism. A MAC frame contains the following fields:

- Sender MAC address.
- Receiver MAC address.
- Sender frame sequence number.
- Acknowledged frame sequence number.
- CRC checksum.
- Payload.

A MAC frame is placed in the payload field of a PHY frame so that it can be sent/received through PHY layer.

### 2.2.1 Sequence Number and Error Detection

The sequence number and CRC checksum fields are used to drop out-of-order packets and corrupted packets. A sliding window transmission control is implemented to provide reliable in-order data transmission.

### 2.2.2 Round Robin MAC

### 2.2.3 Time-division multiplexing MAC

## 2.3 CSMA

# 3 Network Layer

Network layer moves one steps further by enabling data transmission between Atherneth and Internet. The IP address within Atherneth are manually configured and each node holds

## 3.1 Internet Protocol

IP is the thin waist of the Internet. It uses a hop-by-hop routing .

### 3.1.1 IP server: the daemon process that sends/receives IP packets

### 3.1.2 IP accessor: the API for accessing network layer service

## 3.2 Network address translation

# 4 Transport Layer

## 4.1 Internet Control Message Protocol and User Datagram Protocol

Both ICMP and UDP are simple and stateless; therefore, they are implemented in the same manner. For the sending side, pack the bytes into the corresponding ICMP/UDP packet, and send them through an IP accessor. They will then be packed into the IP packet and forwarded to the network layer. For the receiving side, they wait until the next packet is sent from the IP accessor, unpack them according to the protocol and read out the bytes.

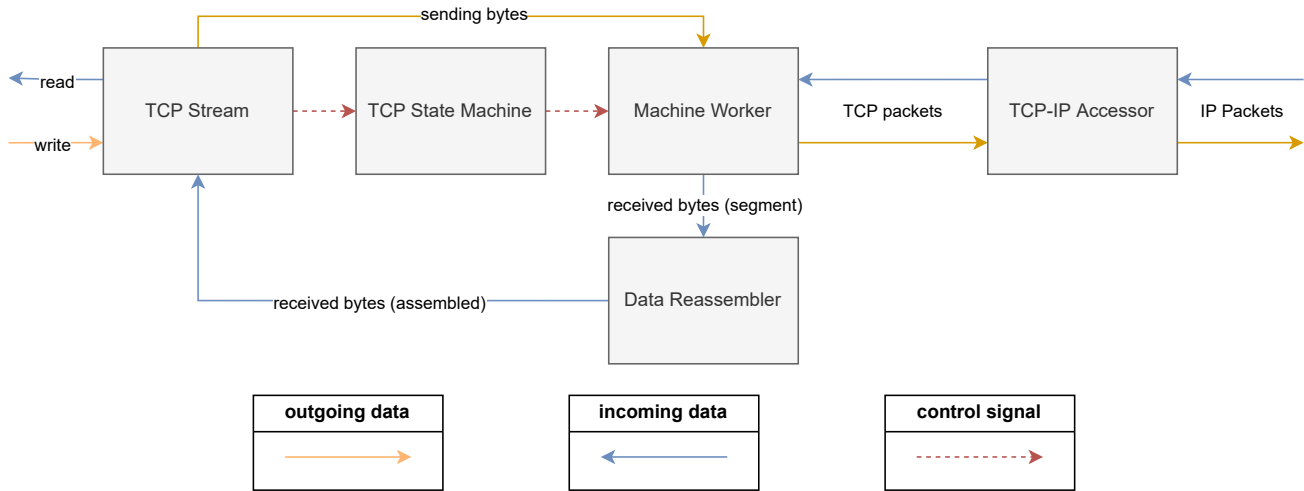


Figure 2: the structure of TCP Stream

## 4.2 Transmission Control Protocol

TCP has been divided into two parts: *TCP Stream* and *TCP Listener*. TCP Stream represents a single TCP connection between the local and remote socket. After creating a TCP stream by either connecting to a remote host or accepting a connection on a TCP Listener, data can be transmitted by reading and writing to it. A TCP Listener is a TCP server listening on a specified port for incoming connections. After accepting an incoming connection, it will produce a TCP Stream.

### 4.2.1 TCP Stream

As depicted in Figure 2, TCP Stream is composed of five parts.

- **TCP Stream:** The host interacts directly with the TCP Stream. TCP Stream will send the data to the Machine Worker and receive data from Data Reassembler.
- **TCP State Machine** will handle the control signal (such as connecting to the remote or closing the connection) from TCP Stream and send them to the Machine Worker. TCP State Machine is also responsible for releasing the resources after the host drops TCP Stream.
- **Machine Worker** handles the state transition of the TCP. It will also pack the data from the TCP stream, send them to the TCP-IP Accessor, unpack the incoming TCP packets, and send the unpacked bytes to Data Reassembler.
- **TCP-IP Accessor** is the bridge between the network layer and the transport layer. It packs the TCP packets into IP Packets, periodically receives IP packets, and sends the unpacked TCP packets to the Machine Worker.
- **Data Reassembler** receives the byte segment from the Machine Worker and reassembles them according to the TCP sequence number. The reassembled bytes will then be sent to the TCP Stream for the host to read.

### 4.2.2 TCP Listener

As shown in Figure 3, TCP Listener is composed of three parts:

- **TCP Listener** responds to the host's request. It will ask the Listener Worker for information on the incoming connection if it tries to accept a connection. The created TCP Stream will change its TCP-IP accessor to the Dispatcher.
- **Listener Worker** receives the connection from the Dispatcher and temporarily stores it for the TCP Listener.
- **Dispatcher** owns an LTCP-IP Accessor to communicate with the IP server. On receiving the IP packet, it will dispatch the inner TCP packet according to its Address map. If the TCP is a SYN packet, it will create a new entry in the Address Map and send the information to the Listener Worker. On receiving TCP packets from the TCP Stream, it will pack them into IP packets and forward them to the IP server.

The TCP Listener is capable of handling multiple connections at the same time.

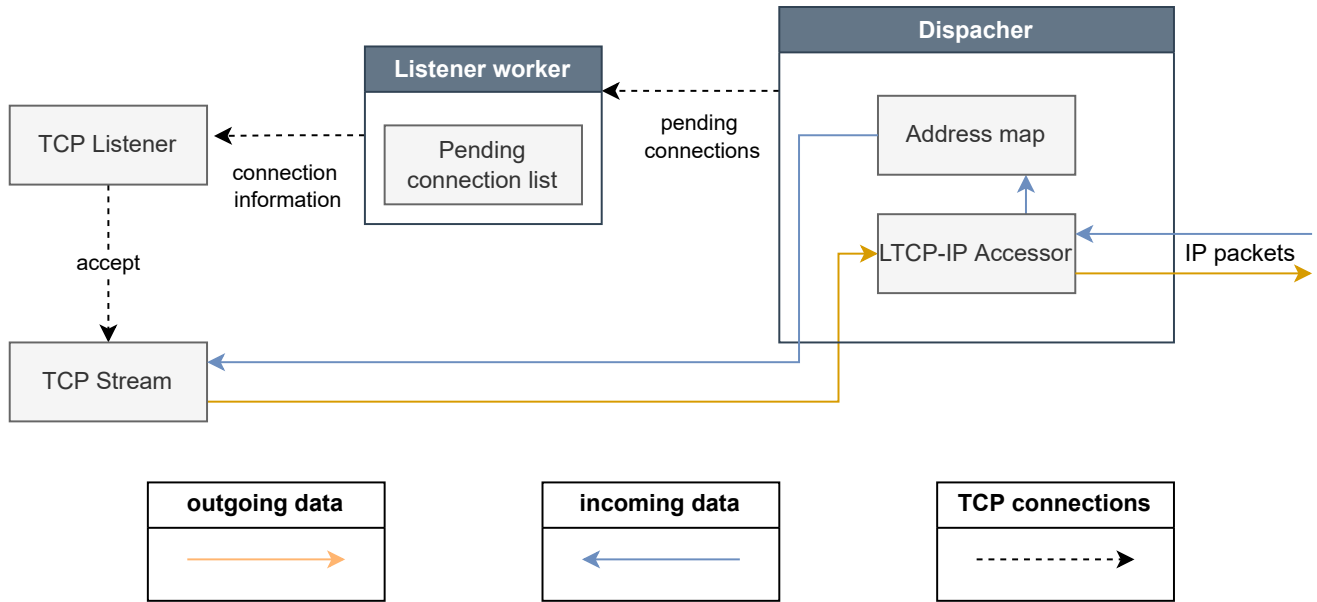


Figure 3: the structure of TCP Listener

## 5 Application Layer

Curabitur tellus magna, porttitor a, commodo a, commodo in, tortor. Donec interdum. Praesent scelerisque. Maecenas posuere sodales odio. Vivamus metus lacus, varius quis, imperdiet quis, rhoncus a, turpis. Etiam ligula arcu, elementum a, venenatis quis, sollicitudin sed, metus. Donec nunc pede, tincidunt in, venenatis vitae, faucibus vel, nibh. Pellentesque wisi. Nullam malesuada. Morbi ut tellus ut pede tincidunt porta. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam congue neque id dolor.

### 5.1 File Transfer Protocol

Donec et nisl at wisi luctus bibendum. Nam interdum tellus ac libero. Sed sem justo, laoreet vitae, fringilla at, adipiscing ut, nibh. Maecenas non sem quis tortor eleifend fermentum. Etiam id tortor ac mauris porta vulputate. Integer porta neque vitae massa. Maecenas tempus libero a libero posuere dictum. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aenean quis mauris sed elit commodo placerat. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Vivamus rhoncus tincidunt libero. Etiam elementum pretium justo. Vivamus est. Morbi a tellus eget pede tristique commodo. Nulla nisl. Vestibulum sed nisl eu sapien cursus rutrum.

### 5.2 Tunneling

Nulla non mauris vitae wisi posuere convallis. Sed eu nulla nec eros scelerisque pharetra. Nullam varius. Etiam dignissim elementum metus. Vestibulum faucibus, metus sit amet mattis rhoncus, sapien dui laoreet odio, nec ultricies nibh augue a enim. Fusce in ligula. Quisque at magna et nulla commodo consequat. Proin accumsan imperdiet sem. Nunc porta. Donec feugiat mi at justo. Phasellus facilisis ipsum quis ante. In ac elit eget ipsum pharetra faucibus. Maecenas viverra nulla in massa.

## 6 Conclusion

Nulla ac nisl. Nullam urna nulla, ullamcorper in, interdum sit amet, gravida ut, risus. Aenean ac enim. In luctus. Phasellus eu quam vitae turpis viverra pellentesque. Duis feugiat felis ut enim. Phasellus pharetra, sem id porttitor sodales, magna nunc aliquet nibh, nec blandit nisl mauris at pede. Suspendisse risus risus, lobortis eget, semper at, imperdiet sit amet, quam. Quisque scelerisque dapibus nibh. Nam enim. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc ut metus. Ut metus justo, auctor at, ultrices eu, sagittis ut, purus. Aliquam aliquam.

## Acknowledgement

Etiam suscipit aliquam arcu. Aliquam sit amet est ac purus bibendum congue. Sed in eros. Morbi non orci. Pellentesque mattis lacinia elit. Fusce molestie velit in ligula. Nullam et orci vitae nibh vulputate auctor. Aliquam eget purus. Nulla auctor wisi sed ipsum. Morbi porttitor tellus ac enim. Fusce ornare. Proin ipsum enim, tincidunt in, ornare venenatis, molestie a,

augue. Donec vel pede in lacus sagittis porta. Sed hendrerit ipsum quis nisl. Suspendisse quis massa ac nibh pretium cursus. Sed sodales. Nam eu neque quis pede dignissim ornare. Maecenas eu purus ac urna tincidunt congue.

## Reference

Donec et nisl id sapien blandit mattis. Aenean dictum odio sit amet risus. Morbi purus. Nulla a est sit amet purus venenatis iaculis. Vivamus viverra purus vel magna. Donec in justo sed odio malesuada dapibus. Nunc ultrices aliquam nunc. Vivamus facilisis pellentesque velit. Nulla nunc velit, vulputate dapibus, vulputate id, mattis ac, justo. Nam mattis elit dapibus purus. Quisque enim risus, congue non, elementum ut, mattis quis, sem. Quisque elit.