计算机网络作业 3-1
利用 udp 在用户空间实现面向连接的可靠数据传输

姓名：林语盈

学号：2012174

班级：计算机卓越班

目录

一、 报文段设计 3

二、 握手挥手过程 4

(一) 握手过程 4

(二) 挥手过程 4

三、 差错检测 5

四、 超时重传 6

五、 日志实现 7

一、 报文段设计

表 1: 报文段设计

共 1024 字节					
Byte 0	Byte 1	byte 2	byte 3	byte 4、5	byte 6-1023
校验和	Flags	Seq	Ack	Data 段长度	数据
Head 段					Data 段

在本次实现中，报文段设计如表 1 所示，具体描述约定如下：

1. 所有的报文段格式一致

2. 报文段最长为 1024 个字节，其中头部 6 字节，数据 1018 字节。第一个字节为校验和字段，第二个字节为 flag 字段，第三个字节为 seq 字段，第四个字节为 ack 字段，第五、六个字节为数据段长度字段，第 7-1024 个字节为数据段。一个字节可以表示的值范围是[0,255].

3. flag 字段共 8 位，格式如表 2 所示。

表 2：flag 字段格式

Byte 1				
Bits 0-3	Bits 4	Bits 5	Bits 6	Bits 7
0	FLAG_SYN	FLAG_FIN	FLAG_SEQ	FLAG_ACK

二、 握手挥手过程

（一） 握手过程

握手过程与 tcp 三次握手相同：

1. 首先由发送端发送第一次握手报文，发送 syn 与 seq=j

2. 接收端接收到第一次握手报文后并检验校验和正确、是 syn，才会发送第二次握手报文，否则一直等待。第二次握手报文为 syn、seq=k，ack=j+1

3. 发送端接收到第二次握手报文后才会发送第三次握手报文，如果发送端接收到的第二次握手报文有错误，则重新发送第一次握手报文。报文为 seq=j+1，ack=k+1

4. 接收端接收到第三次握手报文后建立连接，检查校验和、seq、ack，而如果接收端接收到的第三次握手报文有错误， 则回到等待第一次握手报文的状态 。

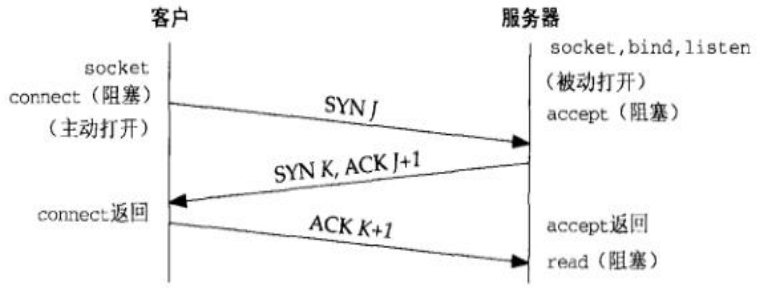

图 1 三次握手状态图

（二） 挥手过程

由于在本实验中，发送端与接收端固定，所以不需要如 tcp 的四次握手，仅需 2 次握手即可。挥手过程如下：

1. 发送端发送数据完毕，并接受到全部 ack 后，首先由发送端发送第一次挥手报文。报文为 FIN，ack=k，seq=j。

2. 接收端接收到第一次挥手报文后才会发送第二次挥手报文并断开连接，否则一直等待 。第二次挥手报文为 FIN、ack=j+1，seq=k。

3. 发送端接收到第二次挥手报文后才会断开连接。（另：实际上，总存在最后一次的 ack 数据包，若其丢失，无论几次挥手都是无解的。）

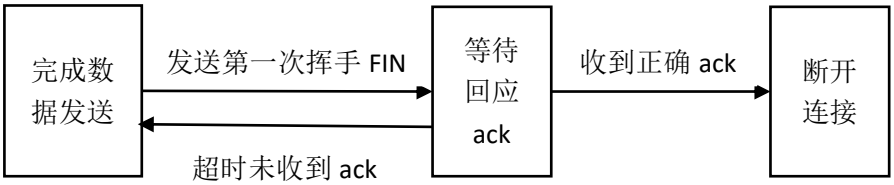

图 2.1 发送端挥手状态图

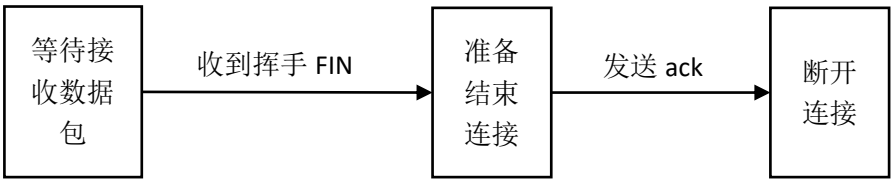

图 2.2 接收端挥手状态图

图 2 挥手状态图

三、 差错检测

1. 发送方：利用 udp 校验和进行差错检测，按 8bits 二进制反码求和运算，计算结果为 8bits，将计算结果取反写入校验和域段（即数据报的第一个字节）。

2. 接收方：计算包括校验字段的反码求和，若和为全 1，则没有错误，否则有错。

校验和算法

```
unsigned char cksum(char* flag, int len) {//计算校验和
    if (len == 0) {
        return ~(0);
    }
    unsigned int ret = 0;
```

```
    int i = 0;
    while (len--) {
        ret += (unsigned char)flag[i++];
        if (ret & 0xFF00) {
            ret &= 0x00FF;
            ret++;
        }
    }
    return ~(ret & 0x00FF);
}
```

四、 超时重传

1. 实现的是非阻塞的超时重传

2. 发送端首先将待发送的数据分成多个包，然后依次发送。每个包拥有不同的 seq（0-255），当发送一个包后，发送端等待返回同样 seq 的 ack，若超时未收到对应 ack 或检查校验和错误，则重新发送这个包。直到收到对应 ack，则继续发送下一个包（停等机制）。

3. 接收端收到一个包后，检查校验和、是否为新的数据包，若错误则发送上一次的 ack；若均正确，根据长度字段读取 data，并发送与 seq 相同的 ack。此外，如果收到的包包含 FLAG_FIN，表示对方请求挥手，则终端连接，返回 FIN 与 ACK。

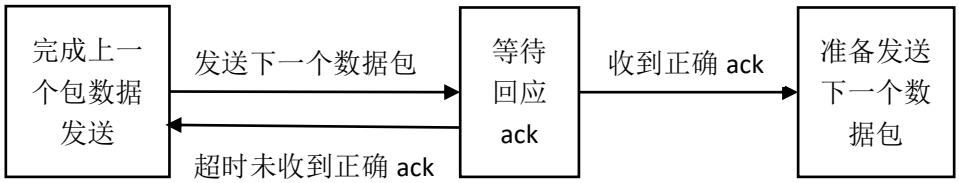

图 3.1 发送端数据传输状态图

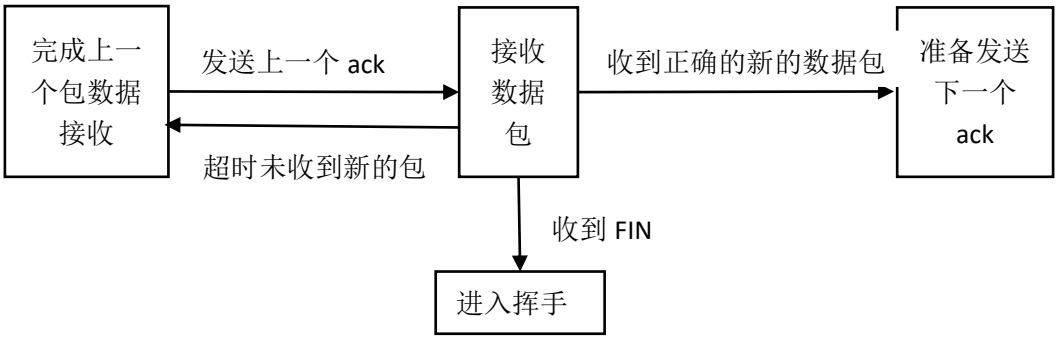

图 3.2 接收端数据传输状态图

图 3 数据传输状态图

五、 日志实现

1. 日志格式为：发送类型（例：第一次挥手），flags，序列号，ack，校验和，报文长度字节数

2. 每次发送和接收数据报都会输出日志

3. 在数据报发送和接收全部完成时利用总字节数和传输时间计算平均吞吐率

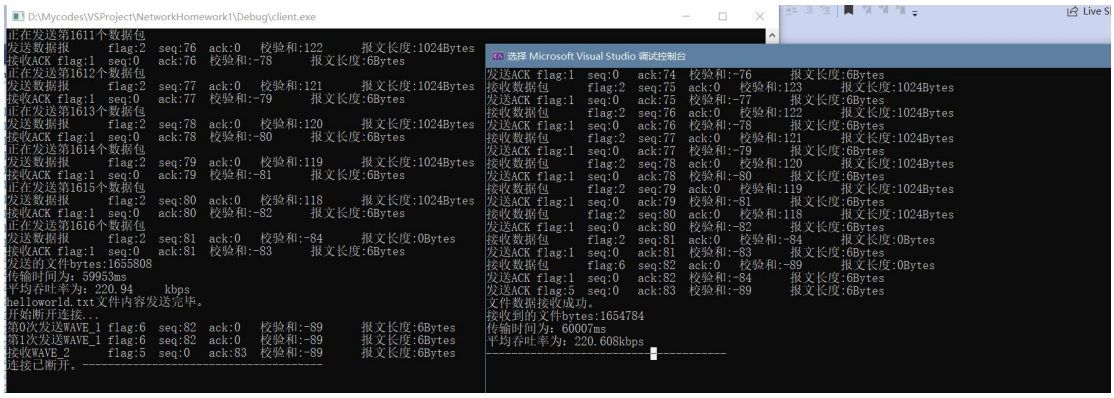

图 4: 日志实现效果图