**Pseudocode**

**Server**

Server Start:

Initialize Variables

Bind to a Port

Read Client:

Loop:

Read data from client

Handshake:

If SYN send SYNACK, Wait for ACK

If ACK, or SEND, complete handshake

If SYN, resend SYNACK

If data is SEND [filename] [length of file]

Create Connection

If file already exists

Open file

Else create new file

While data is less than length

Read data from create socket

ACK data received from create socket

If FIN received or connection timeout, end read

Write data to file

Close file

Close socket

If data is CLOSE

Close the connection

If data is EXIT and there is only one connection

Close the connection

Close the server

Create Connection:

Initialize Variables

Create a Socket

Bind to Port

Connect to client

**Client**

Client Start:

Create Process for Server (Client)

Initialize Variables

Create the Socket

Bind to Port

Connect to Server (Server)

Get user input

If input is SEND [filename]

Get the file length

Send SYN to server, wait for SYNACK

If SYNACK send ACK

If timeout waiting for SYNACK, resend SYN

Tell Server (client) it’s SEND [filename] [filelength]

Start the client server

Send first half of window, wait for ACK

If ACK is earliest send, slide window up

If next packet has been ACK’d as well, keep sliding window up

If ACK timeout, resend data

If all data sent and all packets ACK’d, send FIN

If FINACK received, end sending packets

If input is CLOSE

Send CLOSE to server(server)

Close server(client)

End Program

If input is EXIT

Send EXIT to server(server)

Close server(client)

End Program

**Router:**

Create process for Server (Router):

Initialize Variables

Create a Socket

Bind to a Port

Call Listen to Listen for connections

Listen:

Loop:

Accept a connection

If packet is from Client

Send to Server

If packet is from Server

Send to Client

If value of drop rate is greater than 0

Generate number

If generated number is less than or equal to passed value, drop packet

If value of delay is greater than 0

Hold packet for current time plus delay value

Send packet after expiry time is reached

State Diagrams: