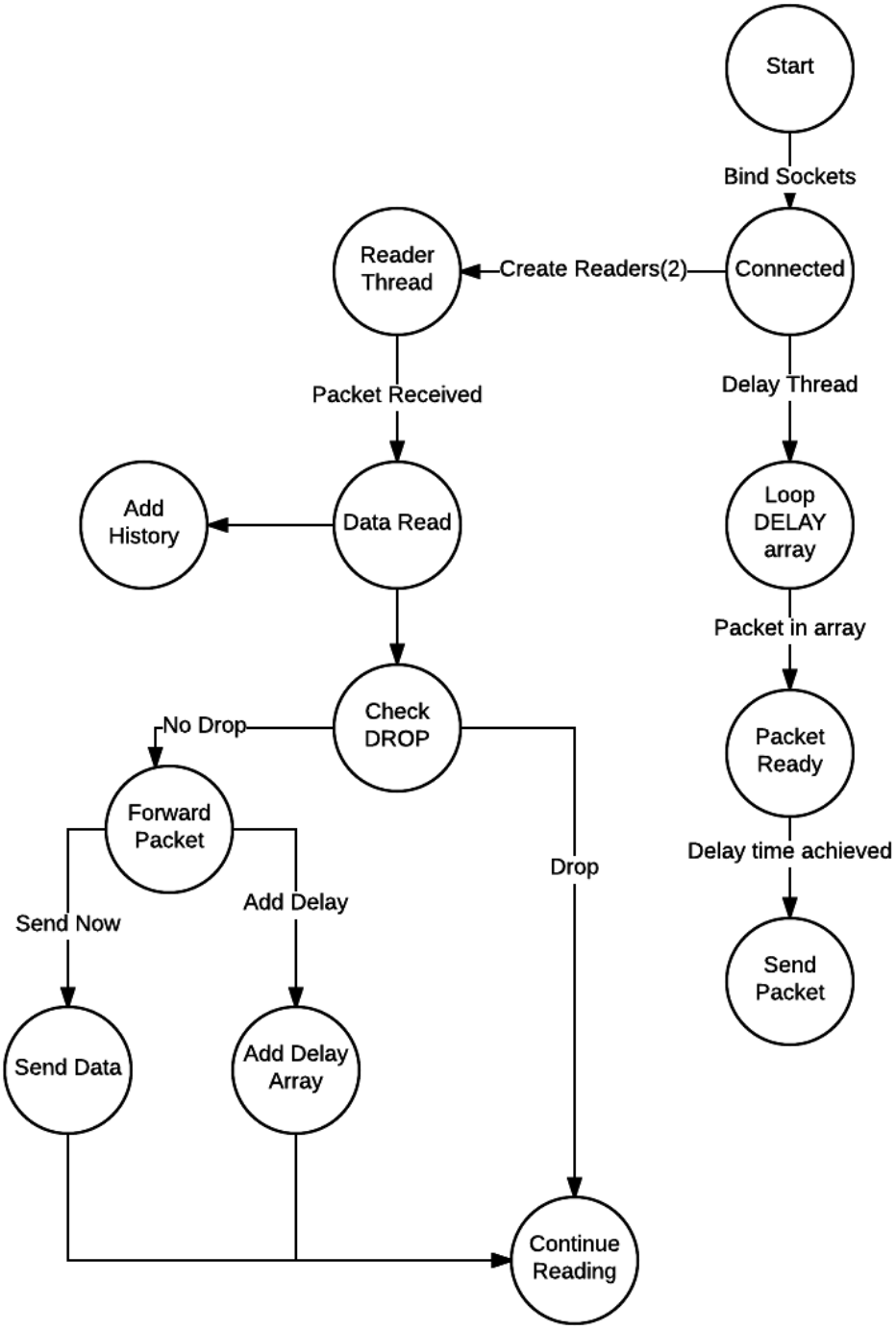
State Diagrams and Pseudocode

Emulator



Start

Start the program

Connected

Connect Pressed
Parse address and ports
Bind sockets
Create READER threads x2 – 1 for each client
Create DELAY thread

Reader Thread

Read Socket in loop

Data Read

Packet received
Check DROP packet random
If DROP
    Continue Reading
If NO DROP
    Forward Packet

Check Drop

Randomize number
IF less than drop rate
    Drop Packet

Add History

Parse Packet type, ackNum, synNum
Create Particle
Set Particle Information
Add to History Array

Forward Packet

If ADDING DELAY
    Add packet to delay array
If NO DELAY
    Send Data to opposite client
Continue Reading

Send Data

Send packet to opposite client

Add Delay Array

Randomize delay
Add packet to delay array

Continue Reading

Data receive complete, continue read loop

Loop DELAY Array

Loop through array
If Delay Entry exists
If delay complete
PACKET READ

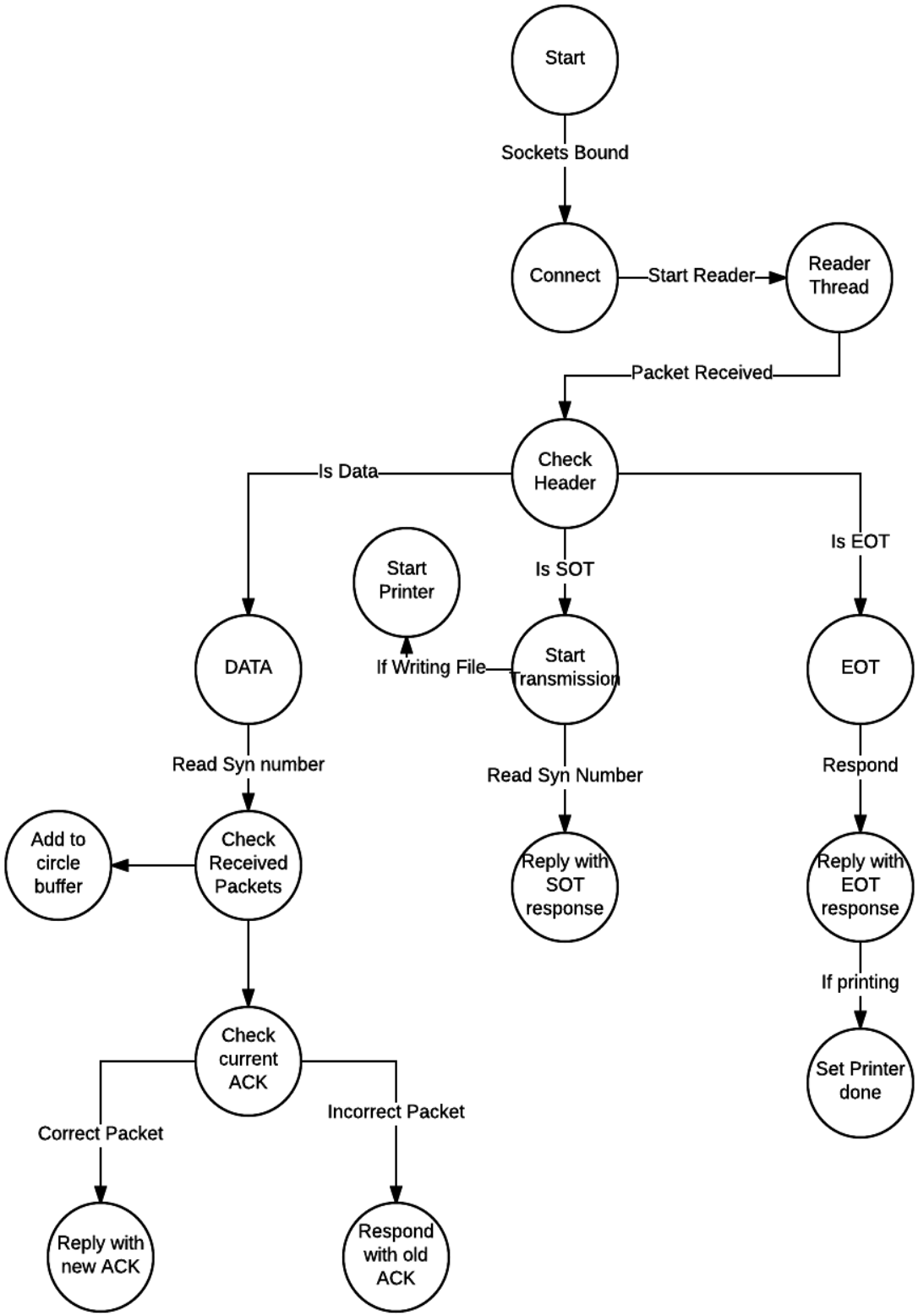
Packet Ready

Delay met – packet ready to be sent

Send Packet

Send packet to opposite client

Client – Reading Side



Start

Start Program

Connect

Bind Sockets
Create Reader Thread

Reader Thread

Loop reading socket
If packet received
    Check Header

Check Header

Check Header Bits
If Data
    GO TO DATA
If Start of transmission
    GO TO Start Transmission
If EOT
    GO TO EOT

Data

Data packet received – read SYN number
GO TO Check Received Packets

Check Received Packets

If current packet is NEW PACKET
    Add to circle buffer
    Add to received packets buffer

GO TO Check current ACK

Add to Circle Buffer

Insert into buffer based on SYN number
Loop buffer array from SYN index
    If buffer array set
        Increase HEAD
    If buffer not set
        Return

Check Current ACK

If SYN matches requested ACK
    Loop received packet buffer – set ACK to furthest packet found
    GO TO Reply with NEW ACK

ELSE

GO TO Respond with OLD ACK

Reply with NEW ACK

Send ACK packet to channel

Reply with OLD ACK

Send ACK packet to old channel

Start Transmission
- SOT header detected
- Read SYN number
- GO TO Reply with SOT Response
- IF Writing Transfer to File
  - START PRINTER

Reply with SOT Response

Respond with SOT Response – ACK number SAME AS SYN

Start Printer

Create Printer thread
Begin printer loop – check circular buffer for new data

EOT

Header bit is EOT
GO TO Reply with EOT Response

Reply with EOT Response

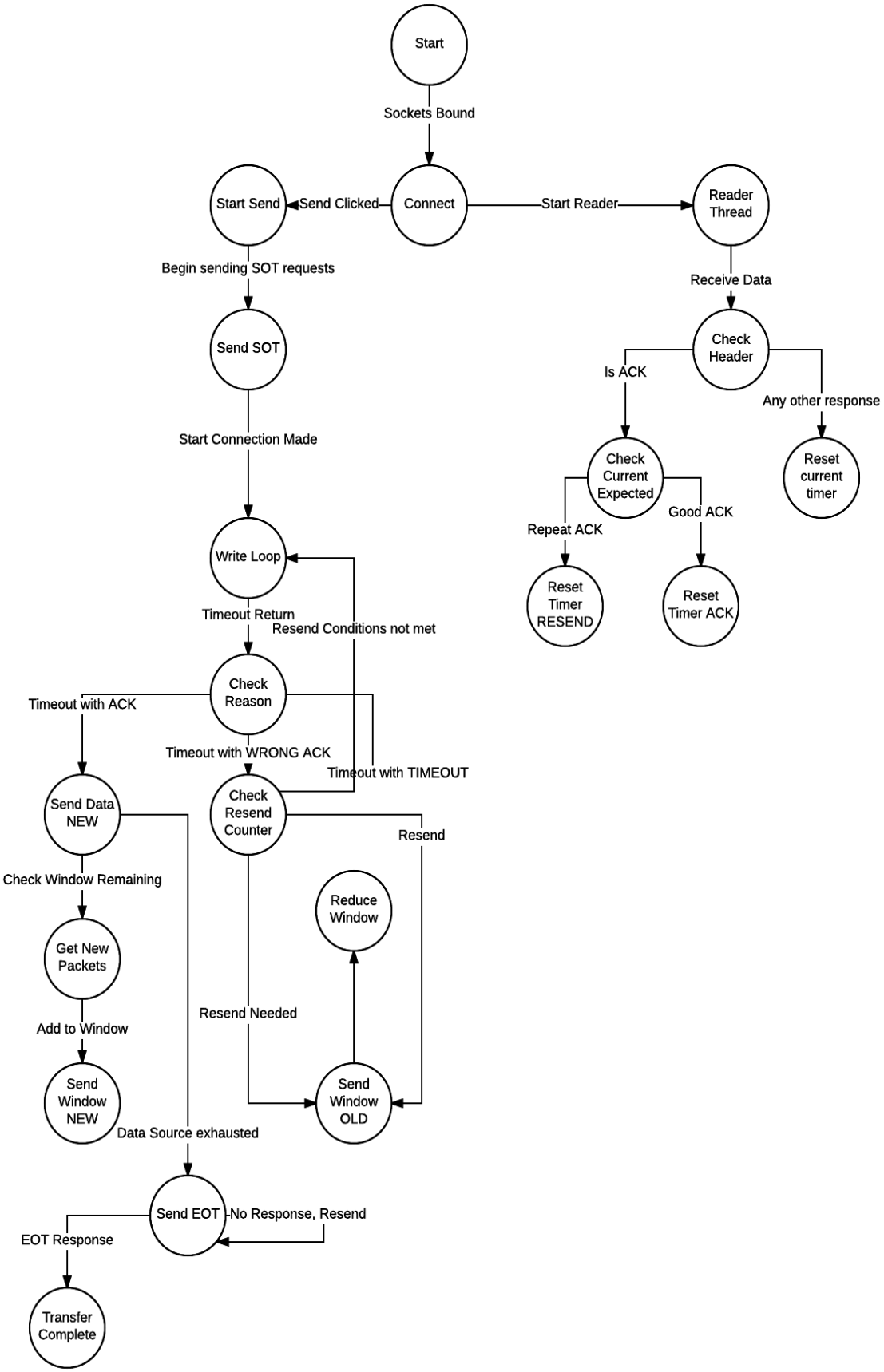
Send EOT Response packet
If Writing File
- GO TO Set Printer Done

Set Printer Done

Break new data checking loop
Close File

Comp7005 Final Assignment
Design Document

Client – Sending Side



Start

Client Started

Connect

Bind receive socket
Set up send socket
Create Read Thread
GO TO Reader thread

Reader Thread

Start Read Loop
IF Data Received
    GO TO Check Header

Check Header

If ACK
    GO TO Check Current Expected
Else
    Reset current time

Check Current Expected

If ACK response > current window index
    GO TO Reset Timer ACK
Else
    Go to Reset Timer RESEND

Reset Timer ACK

Reset send timer with reason ACK

Reset Timer RESEND

Reset send timer with reason WRONGACK

Reset Current Timer

Set timer to the value of header bit found

Start Send

Read Transfer info – packet size, number of packets, window size
Update transfer info
GO TO Send SOT

Send SOT

Loop until max timeouts hit
    Send SOT request to emulator
    Wait on SOT timeout
    IF TIMEOUT
        Increase SOT timeout

IF SOT TIMEOUT > MAX Timeout
Start failed – give up
IF Response
Increment counter
Record RTT
If counter > 3
Average the RTT values from successful SOT
Break loop
GO TO WRITE LOOP

Write Loop

Send initial window
Enter Loop
Wait on timer – first packet in the window
GO TO Check Reason
If transfer complete is set
Break loop, close thread

Check Reason

If ACK Received
GO TO Send Data NEW
IF WRONG ACK Received
GO TO Check Resend Counter
If TIMEOUT
Go to SEND DATA RESEND

Check Resend Counter

If counter > 3
if current time < timer – timer is set after a resend is called and stalls for 0.5RTT
return false – no resend
return true – resend will happen
Reset counter, set timer
ELSE
return false – no resend

Send Data NEW

Check room in window – how much unused
GO TO Get New Packets
Set Timer for each packet: current time + SRTT
If data source is exhausted – GO TO SEND EOT

Get New Packets

Check data source for data
IF data exists
Add to window

Send Window NEW

Check if window has room
Send outstanding packets until window max size met

Send Window Old

Loop from last received ACK pointer in window to end of window – send all packets
GO TO Reduce Window

Reduce Window

Set window size to minimum window

SEND EOT

Loop
    Send EOT
    If EOT Response received
        GO TO TRANSFER COMPLETE
    ELSE
        Resend EOT

Transfer Complete

Transfer Complete
Close File
Print transfer statistics