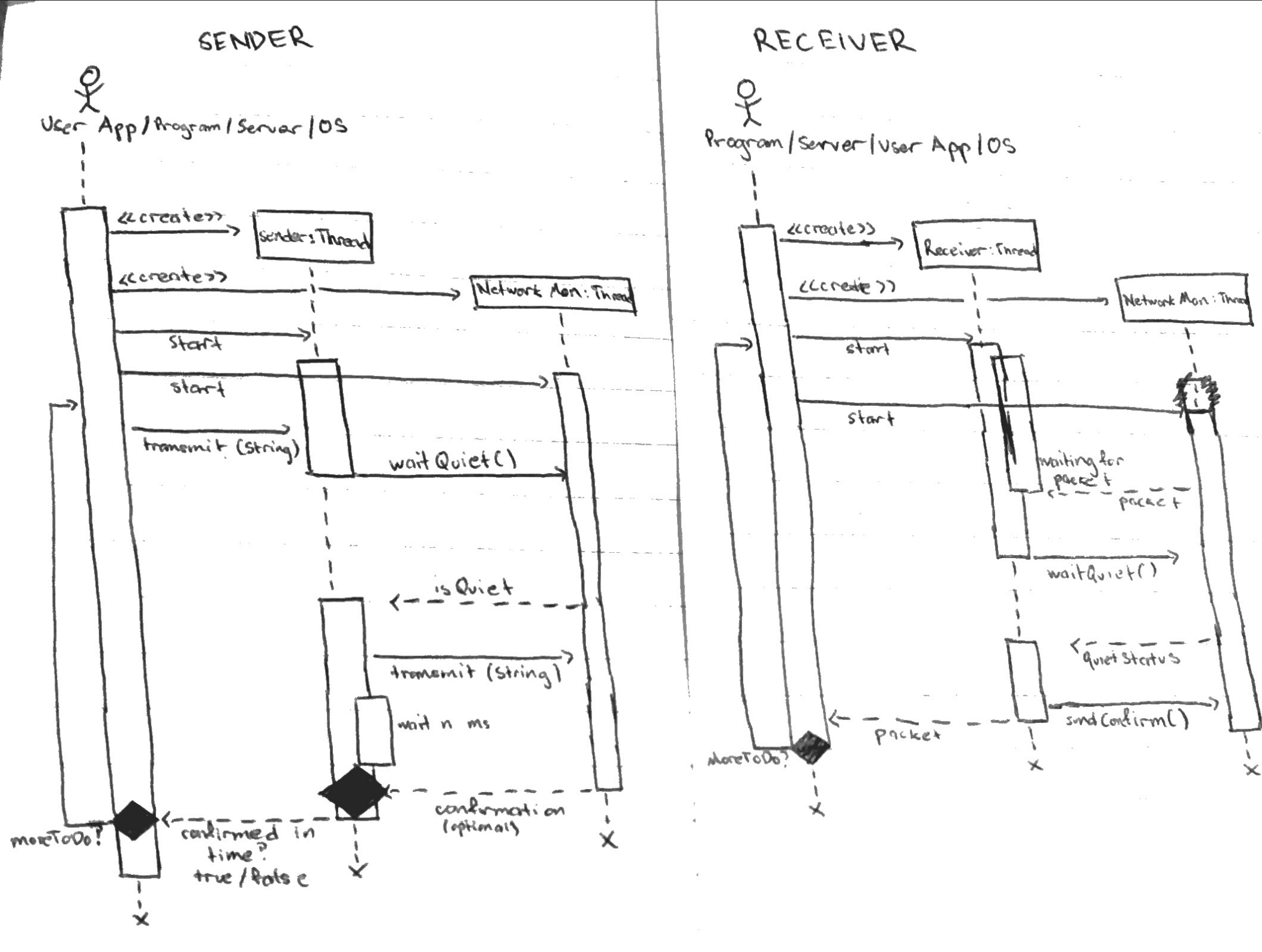
SENDER
User App / Program / Server / OS
<<create>>
Sender: Thread
<<create>>
Network Mon: Thread
start
start
transmit (String)
waitQuiet()
isQuiet
transmit (String)
wait n ms
confirmation (optional)
moreToDo?
confirmed in time? true / false
RECEIVER
Program / Server / User App / OS
<<create>>
Receiver: Thread
<<create>>
Network Mon: Thread
start
start
waiting for packet
packet
waitQuiet()
Quiet Status
sendConfirm()
packet
moreToDo?

* The thread based sequence diagram was chosen because this is somewhat closer to what the OS needs to do (Network card always running)

b) The sender should tell the receiver first how many packets it's going to send, before sending packets. The sender should send all at once when the network is quiet, and then wait for m confirmations. The receiver then accepts each packet, sends a confirmation, receives the next packet, sends another confirmation, etc. This can also be done with one big confirmation, instead of m confirmations. In both cases, the diagram for the sender is altered to send m then wait for m, and the receiver should receive and confirm packets either m times or all at once (latter is more risky for data loss)

2) State diagram