Lab 4: Implementation and study of Stop and Wait Protocols

4.1 Introduction:

The purpose of this experiment is to introduce you to the basics of error correction, time outs and state machines. In this lab, you will be able to provide reliable data transfer between two nodes over an unreliable network using the stop- and-wait protocol.

4.2 Hardware Requirement

- 3PCs with NIU card
- Network Emulation Unit
- Jumper Cables

4.3 Background

Stop and wait is the fundamental technique to provide reliable transfer under unreliable packet delivery system. After transmitting one packet, the sender waits for an acknowledgment (ACK) from the receiver before transmitting the next one. In this way, the sender can recognize that the previous packet is transmitted successfully and we could say "stop-n-wait" guarantees reliable transfer between nodes. To support this feature, the sender keeps a record of each packet it sends. Also, to avoid confusion caused by delayed or duplicated ACKs, "stop-n-wait" send each packet with unique sequence numbers and receive that numbers in each ACK. If the sender doesn't receive ACK for previous sent packet after a certain period of time, the sender times out and retransmits that packet again. There are two cases when the sender doesn't receive ACK; One is when the ACK is lost and the other is when the frame itself is not transmitted. To support this feature, the sender keeps timer per each packet.

4.4 Pre lab questions

1. What are the functions of data link layer?
2. What is the need for flow control techniques?
3. Explain the mechanism of stop and wait flow control technique.
4. In what situations does the sender retransmit a packet?
5. What is meant by promiscuous mode?

4.5 Design

Design a network to illustrate reliable data transfer with packet length of 1000 bytes using stop and wait protocol. Illustrate the performance of the network by changing the time out values from 1000 ms to 4000 ms in steps of 500ms.

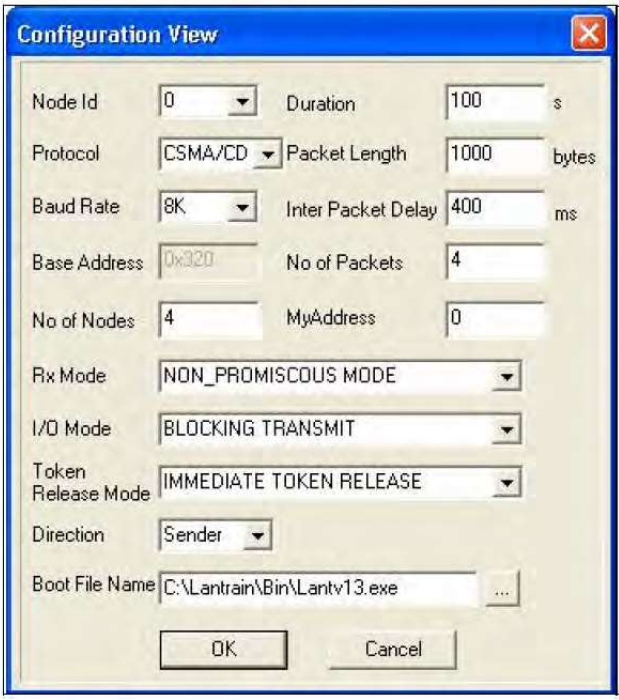
4.6 Procedure:

1. Click on the Stop & Wait icon 

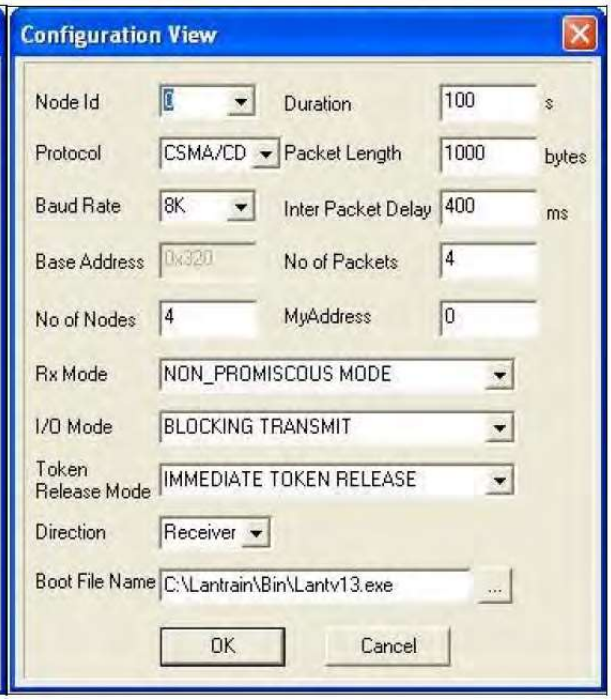
from the desktop on both PCs.

2. Click the Configuration button  in the window in both the Pc's.

PC 1 SENDER



PC 2 RECEIVER



Setting the configuration menu:

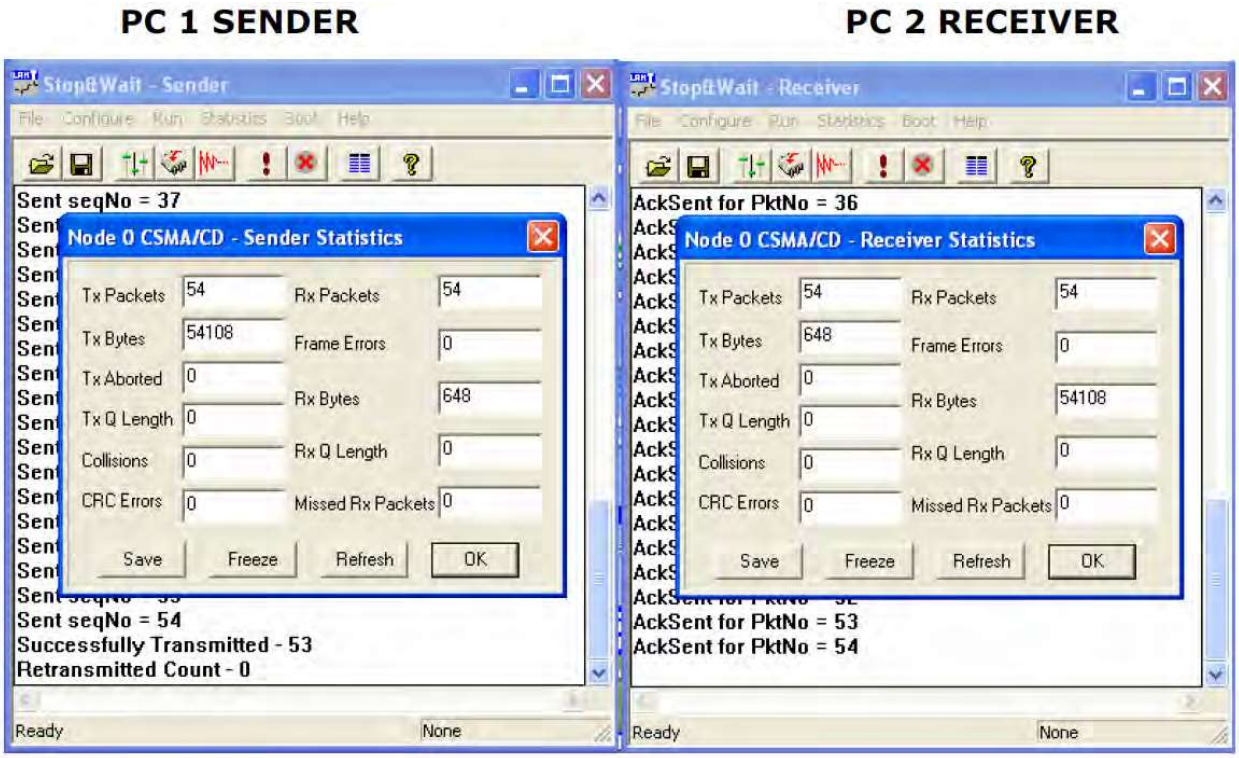
PC 1		PC 2	
Node id	**0**	**Node id**	**0**
Protocol	CSMA/CD	**Protocol**	CSMA/CD
Baud Rate	8Kbps (At both the config menu and NEU)	**Baud Rate**	8Kbps (At both the config menu and NEU)
Duration	100s	**Duration**	100s
Packet Length	1000 bytes	**Packet Length**	1000 bytes
Bit Delay	0(at NEU)	**Bit Delay**	0(at NEU)
Direction	Sender	**Direction**	Receiver

3. Set the Inter Packet Delay to 400msecs

4. Click OK button and Download the driver to the NIU using the BOOT  button command. Booting from any one of the applications is enough.

5. Run the experiment by clicking button  or by choosing RUN _ Start from each application.

6. Set the Timeout Value to 1500 ms



7. Note down the no of successfully Transmitted Packets.

8. Repeat the above steps for various time out values and plot the graph between timeout Value &Throughput. Find the optimum timeout value from the plot.

9. Explain why the throughput is less compared to CSMACD protocol.

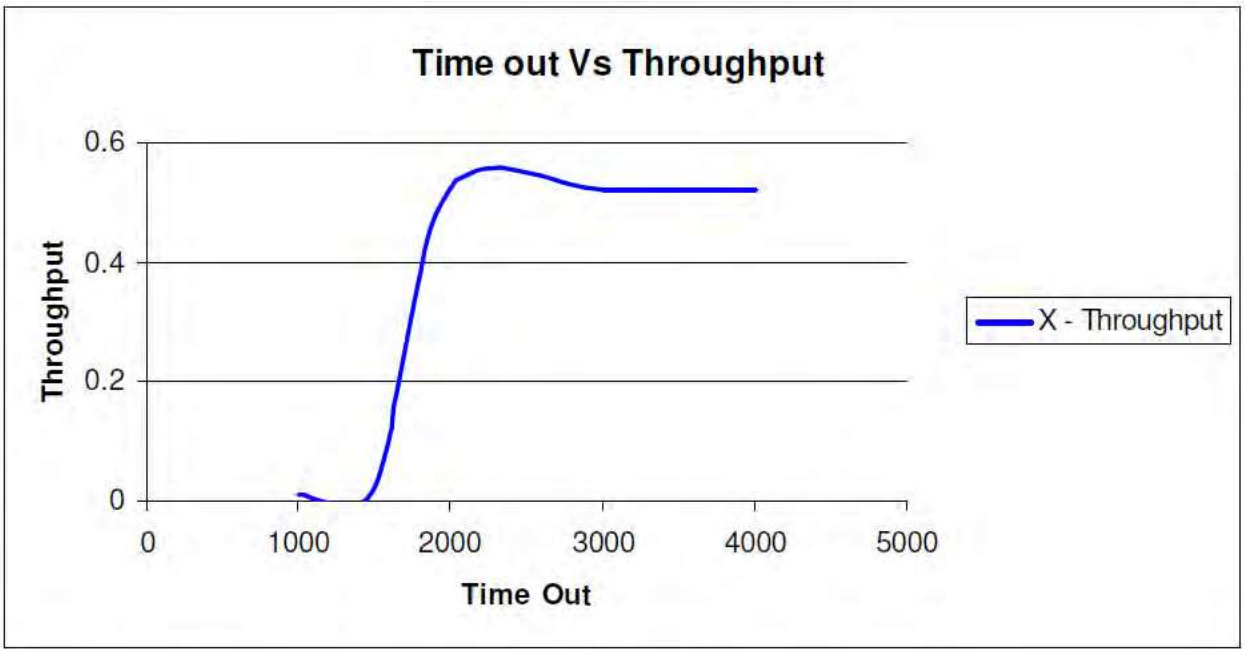
Calculation of Practical Throughput:

$$X = \frac{\text{(Sum of Successfully Tx packets * Packet Length * 8)}}{\text{(Duration of Experiment * Data rate)}}$$

Model Tabulation:

Time out value in ms	Successfully Tx packets	Practical Throughput
1000	1	0.01
1500	2	0.02
2000	52	0.52
3000	52	0.52
4000	52	0.52

Model Graph:



4.7 Post lab questions

1. Computer A uses stop and wait ARQ protocol to send packets to computer B. If the distance between A and B is 4000km, how long does it take computer A to receive acknowledgement for a packet? Use the speed of light for propagation speed and assume the time between receiving and sending the acknowledgement is zero.
2. Why there is no need to number the acknowledgements in stop and wait ARQ?
3. For the question no1 how long it takes for computer A to send out a packet of size 1000bytes if the throughput is 100,000kbps?