CS164 — Fall 1997 Midterm

Aids allowed: Textbook, notes, calculator.

Problem 1. Gigabit Ethernet is developing a standard for transmitting Ethernet frames at a speed of 10^9 bits/sec using all four of the available wires in a twisted pair cable simultaneously.

a The transmitter uses an encoder that selects a combination of voltage levels on the four wires for each byte (i.e., 8 bits) of data. What is the baud rate of this scheme? How many different voltage combinations would be needed to handle all possible data bytes? Could this be done by having the encoder pick one of four possible voltage levels for each of the four wires?

b In addition to the data bytes, the data link protocol must also be able to send a small number (say 10) of "non-data" control symbols, such as idle, start-frame, end-frame, etc. To avoid the need to do bit stuffing at the data link layer, we want to define an additional, unique combination of voltage levels for each of these control symbols. How many voltage levels per wire are needed now?

c Briefly explain how an encoding scheme that uses one combination of voltage levels for each byte of data can cause errors if the data contains long sequences of identical data bytes, for example, the black border on an image.

d Briefly explain how to solve the problem by adding a 9th "clock" bit to the data being sent by the decoder. Does this change increase the number of voltage levels used compared to part (b)?

□

Problem 2. Consider the problem of flow control on the link from Alice to Bob. The transmitting application at Alice wants to download a huge file to Bob, but the receiving application at Bob is sometimes busy with other work. When Bob is busy, he must be able to make Alice slow down or else his receive buffer may overflow and cause packets to be lost. Assume that: (i) the link can carry 1000 fixed-size packets/sec; (ii) the one-way propagation delay from Alice to Bob (or Bob to Alice) is 5 milliseconds; (iii) Alice is always capable of transmitting 1000 packets/sec (PPS); and (iv) Bob can receive either 1000 PPS or 10 PPS according to the following pattern: between time 0 and 4 seconds he can receive 1000 PPS, between 4 and 5 seconds he can receive 10 PPS, between 6 and 9 seconds he can receive 1000 PPS, and so on.

a If they use Stop-and-Wait, what is the maximum throughput that can be achieved in PPS, assuming the length of the ACKs is very short? Suppose Bob returns an ACK as soon as he receives a frame and it passes the CRC. Can his receive buffer overflow under Stop-and-Wait? If not, explain why not. If so, then when should he return the ACK?

b Ignoring errors, is a sliding window algorithm with a transmit window size of 20 packets large enough to allow Alice to transmit at full speed when Bob is not busy? Explain.

c During each 5 second period, Bob is capable of receiving a maximum of 1000 PPS for 4 seconds and 10 PPS for 1 second, for a total of 4010 packets over 5 seconds, or an average of 802 PPS. Thus, Alice could transmit at a constant rate of 800 PPS without Bob exceeding the *average* rate at which Bob can receive packets. How large would Bob's receive buffer need to be to avoid dropping packets in this case?

d Suppose Alice and Bob are using a sliding window algorithm with a transmit window size of 20 packets. Bob wants to delay the ACKs in order to force Alice to slow down when he is busy with other work. What is the minimum size for Bob's receive buffer if he never wants to make sure he never has to drop any packets because of buffer overflow? Briefly describe the algorithm he would use to decide when to return an ACK. Your algorithm should allow them to transfer data at the maximum rate of 802 PPS.

□

Problem 3. UCR is about 12 hops away from Cal State San Bernardino via the Internet, but only about 6 hops from Berkeley. Briefly explain how geographical distances and network distances can be so different. How useful would it be to reassign IP addresses to all the networks in the world based on their geographical locations (like telephone area codes)? □