1. **Server Bandwidth**

Consider a server with direct memory access (DMA) in and out of main memory. Assume the server's I/O bus speed is 400Mbps and the memory bandwidth is 850Mbps.

   a. How many switched 2Mbps T1 links could be supported by the server?
   b. Suppose the server switching time is such that it can forward packets at the rate of 1000 packets per second. Determine the throughput as a function of the packet size. Assume the packet size is $L$ bits.
   c. At what packet size does the I/O bus speed become the limiting factor?

**Solution**

   a. The I/O bus speed is slower than memory, so the I/O bus is the bottleneck. A transfer into memory and a transfer out of memory is needed per packet for DMA architecture, each of which crosses the I/O bus. The server can then transfer data at $400/2 = 200$ Mbps. Therefore, $\left\lfloor \frac{200}{2} \right\rfloor = 100$ interfaces can be supported.
   b. Let the length of the packet be $L$ bits. The throughput is then: $\min\{200 \times 10^6, 1000 \times L\}$ bps.
   c. The I/O bus bandwidth becomes the limiting factor when: $1000 \times L = 200 \times 10^6, L = \frac{200 \times 10^6}{1000} = 200$ Kb.

2. **Virtual Circuits**

Consider the use of virtual circuits with the network shown below. Assume that each switch port has an associated variable, NextVCIOut. NextVCIOut is initially set to 0. When an outgoing connection is made through a port, it is assigned $VCI = \text{NextVCIOut}$. NextVCIOut is then incremented by 1.

![Network Diagram]

The following connections are made, in order:

- A connects to G
- A connects to C
- E connects to I
- D connects to F
- G connects to I
- B connects to D

   a. After the connections are made, what is the Virtual Circuit table at each switch? Use the form of P&D Table 3.3, page 177, to report your answer.
   b. What is NextVCIOut for each port at each switch?
   c. What sequence of VCIs does a packet from E to I get?
   d. What sequence of VCIs does a packet from B to D get?

**Solution**

   a.
b.

<table>
<thead>
<tr>
<th></th>
<th>S1</th>
<th>S2</th>
<th>S3</th>
<th>S4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

c. \(E \rightarrow I: 0, 0, 0, 0, 0, 0\) (or \(0, 0, 0, 0, 0, 0, 0, 0\))

d. \(B \rightarrow D: 0, 2, 0, 0\) (or \(0, 2, 2, 0, 0, 0\))

3. **Spanning Tree Algorithm for Intelligent Bridges**
Suppose the Perlman spanning tree algorithm and the bridge learning algorithm for forwarding are used for the network shown below.
a. Indicate which bridge is root, which ports are root ports (i.e. the preferred port for reaching the root bridge), which bridge is the designated bridge for each LAN, and which ports are designated ports (i.e. the ports that connect some LAN to its given designated bridge) in the table below. Hint: bridges that are not designated bridges for any LAN, and ports that are not either root ports or designated ports do not play a role in the routing of packets. The remaining bridges together with the LANs form a spanning tree.

b. Suppose after the configuration is complete, host Mars attaches to LAN B, host Venus attaches to LAN G and Jupiter attaches to LAN E. Suppose Mars sends a message to Venus, then Venus sends a message to Mars, then Jupiter sends a message to Venus. For each of the three messages, indicate which LANs the message is heard on.

<table>
<thead>
<tr>
<th>Bridge</th>
<th>Root</th>
<th>Root Port</th>
<th>LANs</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B8</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>B9</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Sol

4. Multiple Access

Nodes A and B, are attached via a 1800 m cable, and that they each have one frame of 150 bytes (including all headers and preambles) to send to each other. At time t = 0, both nodes attempt to send. There are 4 repeaters between A and B, and each inserts a 20-bit delay. The transmission rate is 50 Mbps. CSMA/CD with backoff intervals of multiples of 1000 bits is used. After the first collision, A chooses K=0 and B chooses K=1 in the exponential backoff protocol. Ignore the jam signal and the inter-delay prior to sending.

a. If the signal propagation speed is $3 \times 10^8$ m/sec, what is the one-way propagation delay (including repeater delays) between A and B in seconds?
b. Find the time (in seconds) that A's packet is completely delivered to B?
c. Now replace the repeaters with bridges, each of which has a 20-bit processing delay in addition to a store-and-forward delay. If only A has a packet to send, find the time (in seconds), when A's packet is delivered at B?

Sol:

a. Total propagation delay is the sum of propagation time through the wire and delays in all repeaters.
Propagation time = Distance / Propagation Speed
= 1800m / 3 x 10^8 m/s
= 6 µs

Delay in Repeater = Delay in bits / Transmission Rate
= 20 bits / 50 Mbps
= 0.4 µs

Since there are 4 repeaters
Propagation Delay = Propagation time + 4 × Delay in Repeater
= 6 µs + 4 × 0.4 µs
= 7.6 * 10^-6 s

b. Propagation delay of 7.6 µs between A and B is smaller than the backoff interval of 20 µs (= 1000 bits/50 Mbps). With CSMA/CD, when the start of the frame from A arrives at B, B will not send a packet. Therefore, A’s packet will be delivered to B without collision or retry. Then

Latency = Propagation Delay + Transmit Time
= 7.6µs + Size / Bandwidth
= 7.6µs + 150 bytes / 50 Mbps
= 7.6µs + 24µs
= 31.6µs = 31.6 * 10^-6 s

Therefore, A’s packet is completely delivered to B at a time 31.6µs after it begins transmitting it. If we include the time required for the first collision to be detected, which is 1000 bit times, or 20µs, the packet is completely delivered 51.6µs after A began, ignoring the inter-frame delay.

c. Total propagation delay is the sum of propagation time through the wire and delays in all bridges. In each bridge, there is a store-and-forward delay, which is equal to the transmit time, and a 20-bit processing delay.

Propagation time = Distance / Propagation Speed
= 1800m / 3 x 10^8 m/s
= 6 µs

Delay in Bridge = Store-and-forward Delay + Processing Delay
= Transmit Time + Processing Delay
= Size / Bandwidth + Delay in bits / Transmission Rate
= 150 bytes / 50 Mbps + 20 bits / 50 Mbps
= 24µs + 0.4µs
= 24.4µs

There are 4 bridges.

Propagation Delay = Propagation time + 4 × Delay in Bridge
= 6µs + 4 × 24.4µs
= 6 µs + 97.6 µs = 103.6 µs

Total latency becomes

Latency = Propagation Delay + Transmit Time
= 103.6µs + Size / Bandwidth
= 103.6µs + 150 bytes / 50 Mbps
= 103.6µs + 24µs
= 127.6µs = 127.6 * 10^-6 s

Therefore, at time t = 127.6µs, A’s packet is completely delivered to B.

5. Ethernet Timing

This problem is about the Ethernet/IEEE 802.11 access protocol. To be definite, suppose that if a host detects a transmission while it is transmitting a frame, then: (i) if the host has already transmitted the 128 bit preamble, the host stops transmitting the frame and sends a 64 bit jamming sequence; (ii) Else the host finishes transmitting the 128 bit preamble and then sends a 64 bit jamming sequence. For simplicity, assume a collision is detected as soon as an interfering signal first begins to reach a host. Suppose the packets are 1024 bits long, which is the minimum length allowed. Hosts A and B are the only active hosts on a 20 Mbps Ethernet and the propagation time between
them is 20 µs, or 400 bit durations. Suppose A begins transmitting a frame at time t = 0, and just before the beginning of the frame reaches B, B begins sending a frame, and then almost immediately B detects a collision.

a. Does A finish transmitting the frame before it detects that there was a collision? Explain.

b. What time does A finish sending a jamming signal? What time does B finish sending a jamming signal?

c. What time does A first hear an idle channel again? What time does B first hear an idle channel again?

d. Suppose each host next decides to retransmit immediately after hearing the channel idle. After the resulting (second) collision: When does A next hear the channel idle? When does B next hear the channel idle?

e. Suppose after the second collision, A decides to wait 128 bit durations to retransmit (if it hears silence after that long) and B decides to retransmit immediately after hearing a silent channel. Is the transmission of host B successful?

Sol
a. No, A does not finish because it hears B after only 800 of the 128+1024 bits are transmitted.

b. When A detects the collision, it has already finished sending the preamble. Thus, it starts immediately sending the jam signal, which is completed at time:

\[ 40 \mu s + 64 / 20 \mu s = 43.20 \mu s \]

When B detects the collision, it has not finished sending the preamble yet. It has to send 128 bits for the preamble and 64 bits for the jamming signal. Thus, B’s transmission is over at time:

\[ 20 \mu s + 64 / 20 \mu s + 128 / 20 \mu s = 29.6 \mu s \]

c. A hears idle channel after the jamming signal of B has reached, at 29.6µs + 20µs = 49.6µs

B hears idle channel after A’s jamming signal has reached, at 43.20µs + 20µs = 63.20µs

d. If A starts to send at t=49.6µs, its signal arrives at B at (49.6µs + 20µs) = 69.6µs. At this time, B has already sent (69.6 – 63.20) µs x 20 Mbps = 128 bits, which means the preamble has been sent already. Then B will directly send the jamming sequence, which arrives at A at 69.6µs + 20µs + 3.2 µs = 92.8µs

If B starts to send at t = 63.20µs, its bits arrive at A at 83.2µs. At this time, A has already sent more than a preamble. A will directly send the jamming sequence, which arrives at B at 83.2µs + 3.2µs + 20µs = 106.4µs.

e. No. Because B’s signal arrives at A at 106.4µs + 20µs = 126.4µs. At that time, A should have waited for at least (126.4 - 92.8) = 33.6µs, which is 672 bits duration, but instead it waited 128 bit durations and retransmitted.

6. Network Utilities

Read the manual pages for the Unix utilities ip, arp and netstat on your machine. Try running the commands with different options. (Not all options at the same time!)

a. Report the hardware address (6 bytes) of the local eth0 interface for the machine you are logged on.

b. Show your Internet-to-Ethernet address translation tables.

c. Show your eth0 network statistics.

Any reasonable answer would work.

a. 52:54:00:12:35:02

b. (call arp)

<table>
<thead>
<tr>
<th>Address</th>
<th>HWtype</th>
<th>HWAddress</th>
<th>Flags</th>
<th>Mask</th>
<th>Iface</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.2.2</td>
<td>ether</td>
<td>52:54:00:12:35:02</td>
<td>C</td>
<td></td>
<td>eth0</td>
</tr>
</tbody>
</table>

c. netstat –i

<table>
<thead>
<tr>
<th>Iface</th>
<th>MTU</th>
<th>Met</th>
<th>RX-OK</th>
<th>RX-ERR</th>
<th>RX-DRP</th>
<th>RX-OVR</th>
<th>TX-OK</th>
<th>TX-ERR</th>
<th>TX-DRP</th>
<th>TX-OVR</th>
<th>Flg</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth0</td>
<td>1500</td>
<td>0</td>
<td>3924</td>
<td>0</td>
<td>0</td>
<td>2324</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>B</td>
</tr>
</tbody>
</table>