M.I.T. Laboratory for Computer Science

Network Implementation Note
No. 18
July 16, 1979

VERSION TWO LOCAL NET INTERFACE DESIGN CONSIDERATIONS

by J. H. Saltzer

This note records many considerations that arose in the design of the Version Two local network ring interface. It also provides a complete specification of the design. The Version One design, inherited from the University of California at Irvine, is taken as the starting point; its successful operation at L.C.S. and U.C.L.A. is a confirmation of the basic design concepts. Further details of that design may be found in published papers [1,2]. This note explores the ways in which the Version Two design is different or might have been different from Version One; most of these differences, except for operating speed, are simplifications, and therefore thought to be of low risk.

Note that this document has been prepared in parallel with the design and first prototype implementation, and that therefore changes may occur as the version two design is shaken down and brought into production. A later note will summarize any design changes arising from the implementation and shakedown experience.

Why a ring?

The most drastic design change considered was to discard entirely the concept of a ring network, and instead use a passive broadcast cable, following the lead of the Xerox PARC Ethernet [3], on the basis that the Ethernet technology has been field proven, its properties are well understood and adequate for the problem, and there is concern for the reliability of a ring network. The decision to explore the ring approach was reviewed carefully, and reaffirmed on technical grounds, as follows:

1. There was a worry that the large size of the Version One ring interface implementation (some 350 TTL chips) suggested that there is something about a ring network that intrinsically requires more complex logic than an Ethernet. This issue was explored in depth, to understand the reasons for the great disparity in implementation sizes for the Version One ring interface, the A.I. Laboratory Chaosnet interface (which operates on similar principles to the Ethernet), and the Xerox Ethernet interface. Three key reasons were found for the differences in size:

---

a) The Version One ring net was designed with many functions and features intended for use in the U.C. Irvine Distributed Computer System project, and that were not provided in either of the Ethernet-type designs. These features include: 32-bit addresses; ability for an interface to have up to eight different program-settable host names; program-settable masks to determine which bits of each packet address and which bits of each host name should actually be matched; and elaborate redundancy check schemes. All of these features could be options on either an Ether or Ring network, and they have no bearing on the intrinsic level of complexity involved in either technology. When these features are stripped away, the basic communication function appears to be implementable in about the same amount of hardware as does the Ethernet.

b) The Irvine designers had the intention of early commitment to VLSI implementation. As a result, design optimization included little attempt to make efficient use of available TTL chips.

c) Fully one third of the chip count of the Version One interface is a full-duplex direct memory access (DMA) interface for the PDP-11 UNIBUS. This interface is part of the Version One implementation primarily for packaging convenience. This is the "host-specific" part of the network hardware, and different hosts require interfaces of different complexity. The programmed I/O interface of the A.I. LISP machine and the microprogrammed I/O interface of the Xerox Alto both happen to require far less logic to implement than does a UNIBUS DMA. Again, this consideration is independent of the choice of Ether or Ring technology.

After discounting these three considerations there seems to be no significant intrinsic difference in the complexity of implementation of the two approaches, and it appears that a straightforward TTL implementation of a minimum-function ring network should require about the same amount of hardware as an equivalent Ethernet.

2. The Ethernet has a significant analog engineering component, while the ring net is almost entirely a digital design. This difference looks very interesting to explore, because of its possible ramifications in ability to exploit rapidly advancing progress in digital technology and VLSI. To understand this difference, consider that an Ethernet transmitter's signal, though a digital waveform, must be receivable by all receivers on the cable. These receivers are at varying distances from the transmitter and therefore will experience different attenuations and echoes. Similarly each receiver must be able to hear every transmitter--there are N(N-1) such combinations that must work in an N-node network, and the transceiver system must be designed conservatively enough that the worst possible receiver-transmitter placement combination (in terms of echo buildup and attenuation) must deliver acceptable performance. The analog noise level contributed by idle transmitters grows with the number of nodes, though probably less than linearly. Finally, in order for the Ethernet carrier sense feature to work, an active transceiver must be able to notice that it is not the only active one. Thus the receiver

part must be capable of detecting the weakest other transmitter during the interbit times of its own transmitter and distinguishing that other transmitter from its own transmitter's echoes. This set of requirements is not impossible to meet, but very careful analog transmission system engineering is needed. In contrast, the analog component of a ring network repeater is more benign. Any given transmitter sends a signal down a private line to only one receiver. The receiver has one echo environment and one received signal level to cope with. Thus, a relatively simple line driver/line receiver combination can suffice. For this reason, the Ethernet technology is straining to reach the 8-10 Mbit/sec. signalling rate with a 200 node net, while the ring can operate at that speed and scale with a fairly elementary analog system.

While engineering in the analog domain is substantially easier in the ring, in the digital domain the situation reverses. The presence of an active repeater at each node of a ring means that careful measures must be taken to insure that the digital logic does not fail, because a repeater failure anywhere disrupts the entire net. In contrast with the Ethernet, the mechanics that grant access to the ring are entirely digital and rely on a circulating token which, if lost, is detected and recovered by digital logic. Some scheme is needed for coordinating the digital clock of each transmitter with that of the next receiver, in such a way that phase errors are absorbed rather than amplified as the ring is traversed. Thus the ring has a wide variety of engineering problems to solve in the digital engineering domain. This difference in the character of the hard engineering problems of the two technologies offers an exploitation opportunity that may favor the ring network. The recent and projected waves of technology improvement have benefited the digital domain more than the analog, mostly because it is easy to see how to solve problems systematically by increasing digital component count; it often seems to be harder to take systematic advantage of increased numbers of components in the analog domain. A less compelling, but still interesting, possibility is that because of the simple analog transmission system required by the ring, even the line drivers and receivers might be integrable into a future VLSI implementation; it is not clear how to do this for the more exotic transceiver technology of the Ethernet.

3. Electromagnetic compatibility between the net and other physically adjacent electrical equipment is generally easier to engineer with a balanced transmission medium than with an unbalanced one. One of the attractions of the Xerox Ethernet is the ease of attaching to it at any point, which ease relies on the use of coaxial cable, an unbalanced medium. If one tried to use a balanced transmission medium for the Ethernet, it would probably become necessary to install connectors every time a new node is added, and easy attachment virtue would be lost. In addition, it is not clear how one would listen for other active transceivers, since in the most obvious balanced waveform modulation schemes the transmitter runs continuously rather than for half of each bit time. In contrast, the ring network can use shielded twisted pair and take advantage of the simpler EMC environment. At the same time, a passive star arrangement for a ring captures much of the easy attachment property. The passive star is described in detail in a recent paper. [4]

4. An attraction of the Ethernet is the intrinsic high reliability that comes from having a minimum number of active components whose failure can disrupt the net. The most important shared component--the coaxial cable--is completely passive. In contrast, the primary objection to a ring network is the operational fragility of a series string of 100 or more repeaters. However, this fragility appears to be easy to overcome by arranging the ring network in a passive star.

5. Another attraction of the Ethernet is that it is exceptionally easy to install--a single cable is routed through the building, near every office or other location in which a network node might be needed. Actual attachment of nodes can be deferred until the node is required, at which time attachment can be accomplished by clamp-on connectors; attachment does not disrupt network operation. However, hand-in-hand with this convenience goes an associated inconvenience, namely that trouble isolation and first-aid repair cannot easily be centralized. Some kinds of failures will require foot-by-foot inspection of the network and each node attachment, involving access to offices throughout the building. The passive star configuration of the ring network appears to completely overcome this potential problem. Extensive field experience with both kinds of networks is really required to determine which is more effective on day-to-day operational issues such as this and the previous two.

6. There is an intrinsic limitation in the Ethernet approach in its ability to make effective use of higher speed transmission media, such as optical fibers, while maintaining high effective usage of the medium. At the beginning of each packet transmission there is a period when there is a risk of collision: this period is proportional to the length of the transmission medium, since the packet is exposed to collision until its first bit propagates to the farthest transceiver. The length of this exposure is thus fixed by the physical configuration. As the transmission speed increases, the time required to transmit an average size packet decreases, until the packet transmission time becomes as short as the cable propagation time. At that point, most of the advantage of carrier sense is lost and the system becomes an ordinary Aloha channel, with an intrinsic data capacity limit of about 18% of the channel capacity. For a 1 Km. cable, the end-to-end propagation time is typically 4500 nsec. This is comparable to the time required to transmit a 60-byte packet at 100 Mbit/sec. Thus an attempt to build a 100 Mbit/sec. Ethernet might result in an effective performance limit near 20 Mbit/sec. The ring, because it does not use a contention access scheme, does not have any corresponding limiting effect, and thus can be scaled up directly to 100 Mbit/sec. configuration.

7. A second limitation of the Ethernet approach that appears to be very difficult to overcome is that it is not clear how one might additionally take advantage of fiber optic technology. This technology offers the attraction of very high speed, excellent electromagnetic compatibility, avoidance of lightning and ground reference problems, and someday low cost. However, the problems of turning optical fiber into a broadcast medium are formidable. One must invent a satisfactory technique for tapping an optical fiber and detecting a signal without diverting too much optical energy or else the system will not scale up very well in

number of nodes. Yet the same tap must allow introducing a new signal without loss. The new signal needs to propagate in both directions from the transmitter. In contrast, since a ring network uses one-way, point-to-point transmission, replacing the electrical links in a ring network with fiber optic links is quite straightforward.

Considering these various technical arguments, it appears that one cannot make a clear case for either the Ethernet or the ring technologies on a priori grounds, that there are some strong reasons for favoring each, and that practical experience with 100-node ring networks is really required to establish concrete comparisons of reliability and ease of maintenance and reconfiguration in the field. Thus there seems to be substantial technical interest in continuing to develop ring technology.

A related idea that was explored briefly was that of using radio frequency broadcast signalling on coaxial cable as, for example, the Mitre Corporation has done [5]. This approach has two appeals:

a) The same coaxial cable can also carry other radio frequency signals with different purposes, for example cable television. Thus bringing the data network into an office would automatically bring the CATV system there, too.

b) The cable television industry has developed a wide range of modestly priced components, including cable attachment hardware and radio frequency signalling chips that one could exploit.

The radio frequency signalling approach, however, has the same kind of large analog engineering component as does the Ethernet, this time in the form of wide-band linear amplifiers, voltage controlled oscillators, filters, modems, and phase locked loops. Although there are available chips that help perform those functions, in real circuits those chips must be surrounded by additional analog components--capacitors, resistors, transformers, etc. In exploiting cable television industry developments, one misses the opportunity to exploit what may be even more potent (by reason of volume and potential total integration) economic forces in the digital logic area. For this reason, the radio frequency signalling approach was not explored further.

Introduction to the ring design

For purposes of discussion, it is helpful to employ consistent terminology, and the following arbitrary labelling is used here. Each local net interface (LNI) has a transmitter and a receiver, which operate in three modes. In match/repeat mode the LNI repeats the stream of bits while checking each packet for a destination address match. When a packet is noticed that is intended for this node, the LNI switches to copy/repeat mode, in which it continues to repeat the stream of bits to its next neighbor, but it also copies it for transmission to its host computer. The term repeat mode means either match/repeat mode or copy/repeat mode. When a host computer wishes to send a packet, it requests that the LNI switch to originate mode. When data transmission is taking place there is one originating LNI and there is some number of copying LNI's. The originating LNI is responsible for removing its

own data packet from the ring. The Version Two LNI is referred to as V.2.LNI and its components are labelled with similarly derivative tree names.

A fundamental design strategy of this local network is to use high data signalling bandwidth, which is assumed to be cheaply available over the short distance and accessible environment of an office building, as an engineering lever to simplify or reduce the cost of other parts of the design. This engineering lever is most obviously at work in the basic concept of a ring network--there are no routing decisions; every packet travels all the way around the ring to be removed by the originator rather than the target. The design assumes that network usage is comparable to that observed in the Xerox PARC Ethernet [6] in which during the busiest second of the day, the load presented by 100 nodes amounts to perhaps 30% of the network signalling rate.

The design also assumes that in the environment of an office building it is possible to install new cables dedicated to the purpose of the local network, so that impedance, dispersion, line balance, and resistance are predictable and controlled. This assumption means that transmission system design can be relatively simple and that noise can be kept relatively low.

Distributed ring control: overview

Probably the single most difficult design aspect of a ring network is the management of the common communication medium (the series string of repeaters) without introducing central control or a preferred node. Two strategies have been proposed that do not depend on any form of central control, contention control and token control. Contention control in a ring net is similar in spirit to the contention control used in an Ethernet or Aloha network--if no data is passing through a repeater, it tries to originate its message, and collisions with other originators may occur, in which case all back off and try again. Various proposals for contention control for a ring have been explored on paper [7] but because of concern for the rate at which collisions might grow as a ring network grows in size, the Version Two network design does not use contention control. (Some provision does appear in the design for experimenting with this idea, because as will be seen, ring initialization introduces some elements of contention control.)

Instead, the Version Two design uses token control, as did the Version One design. With token control, a unique bit pattern called a token is introduced into the ring, and this pattern continually circulates. To originate a packet, a node waits for the token to come by. It transforms the token so that no one else will recognize it, transmits its packet, and then places a new token in circulation. A node is allowed to send only one packet, which has a maximum length, after which it is required to return the token. This approach guarantees fair access to the ring for all participants; no additional anti-hogging mechanism is required. With a token control system, there is no central management of normal ring use, but one must still devise some way for the token to get started, and for recovery in the case that the token gets lost or transmission errors cause two or more tokens to appear; the trick is again to avoid depending on a preferred central node for initialization and recovery. To achieve this end, the Version Two design (as did Version One) uses a contention strategy for initialization. To minimize

the number of different mechanisms, token-controlled message origination includes elements of ring recovery. Initialization, when needed, is accomplished simply by resorting to contention-controlled message origination. Thus the message origination mechanism carries all three functions: normal operation, recovery, and ring initialization.

One minor additional consideration in a token control system is that the ring must be large enough, when measured in bit-times of delay, for a complete token to fit into the ring. (As will be seen, the token encoding involves a multibit token signal.) Since even a many-node ring may be operated with a scenario in which inactive nodes turn their power off, at off-hours one might find only two or three nodes active and a ring too small to hold the complete token signal. Originate mode includes a test for arrival of the closing token before it has been fully transmitted, and if necessary a shift register is switched in series with the ring to artificially provide enough delay to allow the token to circulate.

In the detailed discussion below of design considerations, ring initialization and design features to aid message origination will be found in many places. A complete description of the distributed ring control design, both the hardware of the LNI and the software that drives it, is deferred to a separate section, after the related design features have all been exposed.

Protocol levels in the ring design

In studying these design considerations, one should keep in mind the larger system of which the local net interface forms a part. A group of physically nearby host computers, each with its own local net interface, are connected into a single ring network. This ring network is in turn interconnected to other local networks that may have identical, similar, or quite different speeds and protocols; the interconnecting gateways are computers capable of "bridging" such differences. Because of the possible diversity in the lowest level data transport mechanism, the LNI design explicitly recognizes the existence of two levels of protocol, a "transport" level and a "higher" level. The transport, or level 0, protocol is the one defined by the local net interface; it is responsible only for movement of data from one interface to another within the immediately connected ring. Other examples of level 0 protocols are SDLC and Ethernet; they are sometimes called "multidrop" protocols, because they are used to manage multiple connections, or "drops," on a single piece of wire. Movement of data across a larger network is the responsibility of the "higher" level protocol, which hosts and gateways are assumed to understand. (This higher level protocol is, of course, likely to have several layers itself.) Examples of higher level protocols are ARPA's TCP/Internet, the Xerox PUP, the ARPANET NCP, and the part of IBM's System Network Architecture above the SDLC level.

The level 0 protocol defined by the Version II LNI is, upon closer inspection, subdivided into three sublevels:

level 0.0 analog signalling level--the representation of bits on the wire.

level 0.1 digital signalling level--the encoding of data into a transmitted bit stream.

level 0.2 packet transport level--the format of data packets accepted for delivery.

These three levels operate quite separately from one another, and have largely distinct design considerations, as will be seen.

In addition, for any given host computer, there is another parallel protocol by which a packet is handed from (to) the host computer to (from) the V.2.LNI. Thus we label

level 0.H host interface level--the way a packet is passed between a host and the V.2.LNI.

The complete level zero protocol sequence then is as follows: a host computer constructs a level 0.2 packet. It wraps the packet in a 0.H protocol to pass it to the V.2.LNI. The V.2.LNI unwraps the packet and rewraps it first in the 0.1 digital signalling protocol and then the 0.0 analog signalling protocol. A copying node unwraps the packet from the 0.0 analog signalling protocol and then the 0.1 digital signalling protocol to reobtain the level 0.2 packet, which it wraps in a 0.H protocol to pass to its host. Finally, the copying host unwraps the packet from the 0.H protocol and looks inside the 0.2 packet for the data, which is in the format of the "higher level" protocol. The 0.H. protocol used by the originating node may be different from the 0.H. protocol used by the copying node.

Level 0.0: Analog signalling level

1. The transmission medium is shielded twisted pair, of the smallest, most flexible construction that proves to handle the required data rate. The choice of twisted pair is made primarily on the grounds of electromagnetic compatibility, since a balanced transmission system is less susceptible to interference from other facilities and is less likely to cause interference to other facilities. It also permits direct coupling to differential receivers, so that each LNI can have its own local ground reference. Small size and flexibility are important to installation ease. The cable should be UL-listed as having adequate fire-resistant and low smoke-producing characteristics for use in false ceilings and floors used as air plenums without a conduit. [8]

2. The transmission coding scheme is a Manchester code operating across the twisted pair. When no data is being transmitted, the voltage across the twisted pair is zero and the voltage between the twisted pair and shield is maintained at the same D.C. value as when transmitting data. This "completely balanced" approach minimizes the effects of line charging and

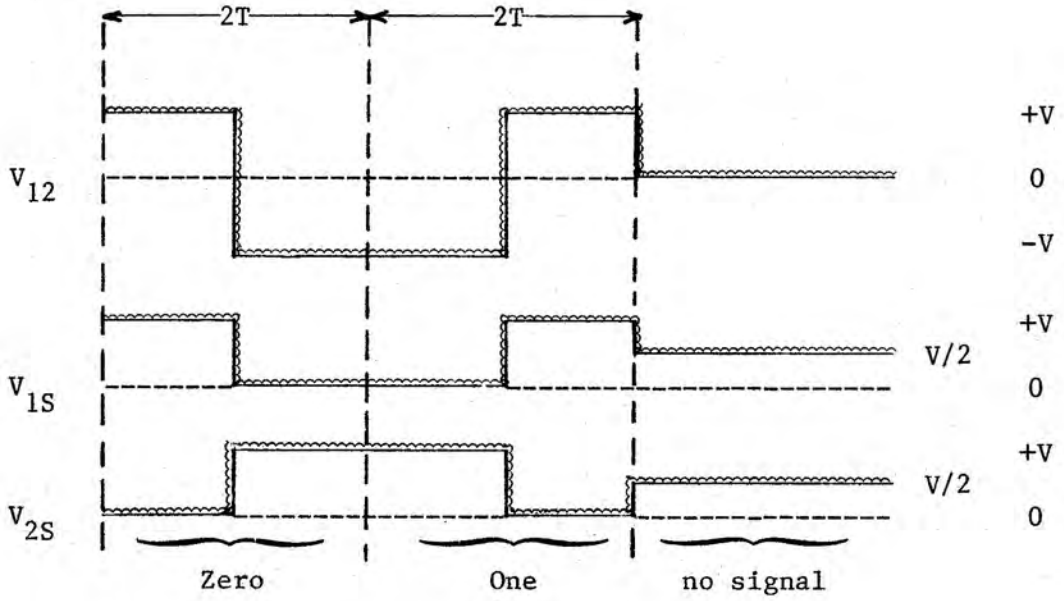
low frequency dispersion. The differential voltage is generated by applying a positive voltage to one or the other of the sides of the twisted pair while grounding the alternate side of the pair. This technique allows all signals to be generated from a single power source. Constant maintainance of an average D.C. level allows a simple transmitter presence detector to be implemented at the receiver or wire center for trouble isolation and it is utilized to provide a "bias" current through the contacts of the LNI bypass relay to insure a low-resistance path through those contacts. Figure one illustrates the voltage levels involved.

3. Each local net interface has its own independent crystal clock, nominally operating at the network standard frequency. This scheme is chosen in preference to one in which the clock of each repeater is derived from the signal it receives, and used to reclock its own transmitter's output, perhaps with a local phase-locked oscillator. (The stability of the latter approach in a closed ring of many repeaters is open to question. Although mathematical analysis suggests that it can be stable, verification in the field with more than a few repeaters has not been carried out anywhere, to our knowledge.)

4. The receiver operates by sampling the differential line voltage at six times the nominal bit rate to watch for transitions. Although a higher sampling rate might appear to be advantageous, the speed limitations of the Schottky TTL logic family would require that more exotic circuitry be employed. The primary reason for choosing the over-sampling technique rather than a transition slope detector or transition-triggered monostable sampler is that both the Xerox Ethernet and the Primenet ring have found it to be successful. An earlier Ethernet design using a monostable sampler was less reliable.

5. The output of the sampler drives a digital filter that removes high frequency noise and provides an estimate of the phase difference between the local clock and that of the transmitting LNI. Whenever the phase difference is observed to have shifted enough that transitions are occurring one sample later (earlier) than before, the receiving local net interface expands (contracts) the next transmitted bit by one-sixth of a bit time, so as to hold the average transmitted bit rate to the proper value despite minor differences in clock rates. The expansion (contraction) is accomplished by lengthening (shortening) both the positive and the negative parts of the next bit equally, so as to maintain line balance. The frequency difference that can be absorbed by this scheme depends on the bandwidth of the filter in the phase estimator. With a high bandwidth phase estimator, a very large frequency difference can be absorbed--as much as 30%. On the other hand, a narrow band phase estimator may be necessary to assure stability. Figure two illustrates.

6. The digital filter and phase difference estimator resets and prepares to resynchronize whenever four or more bit times go by with no transitions observed. This feature is part of the ring initialization mechanism.



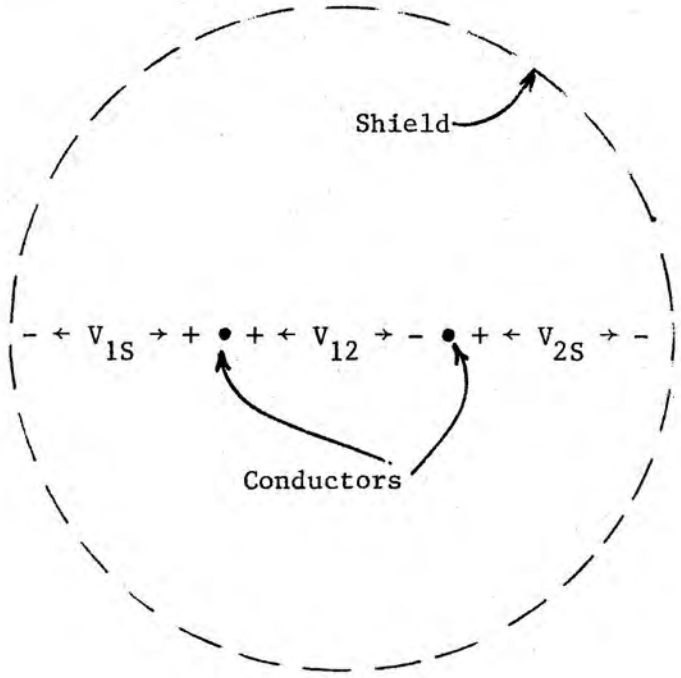


Figure one. Transmitted coding scheme. V is 5.0 volts; for 8.3 Mbits/sec., T is 60 ns.

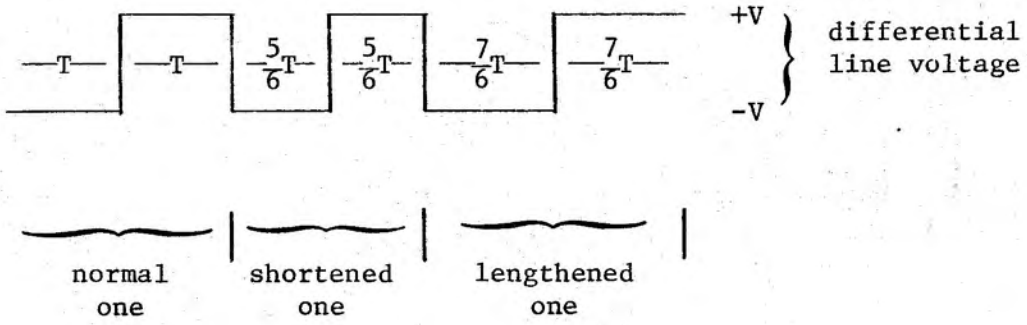


Figure two. Phase correction

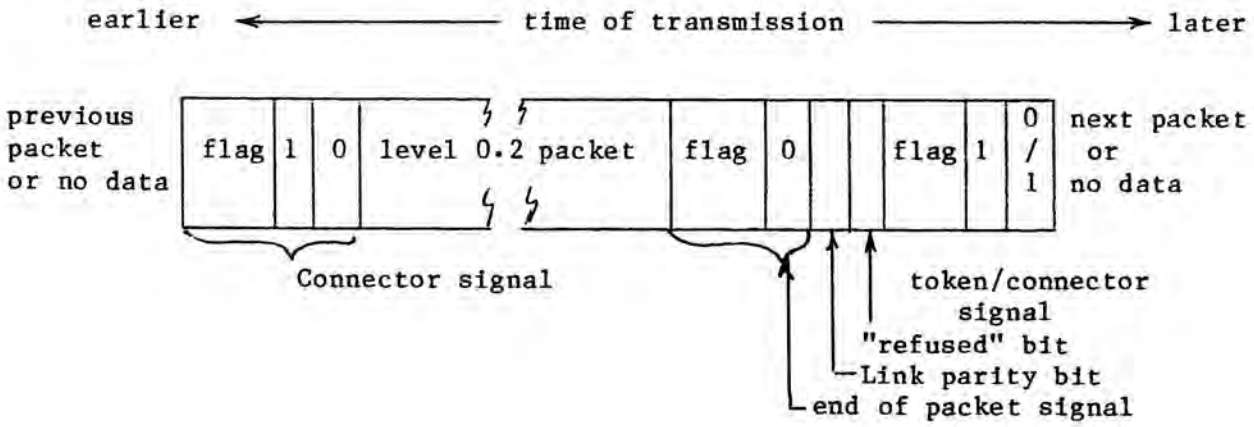
7. The cable shield is connected to the ground reference at the transmitting LNI. At the receiving end the shield is floating. This approach not only avoids ground loops but it also avoids connecting two possibly different ground references together, since each LNI is grounded to its own host computer, which has its own ground system.

8. The reset request from the host has no effect at the analog signalling level. (It is important never to disrupt the repeater.)

9. The Manchester code uses only two out of the possible four two-bit sequences. If either of the other two sequences are observed by a receiver, it is a good indication of trouble on the incoming link or in the adjacent transmitter. As a trouble isolation technique, the receiver reports Manchester code violations by raising the status line "link error" for one subclock time, and it outputs a correctly coded one bit, so as not to raise concern about the health of the next link in the ring. (The alternative of repeating the Manchester code violation has the superficial advantage of passing forward a trouble report as well as an indication that the message may have been damaged, so that the recipient or originator can take action. However, that approach would trigger the "link error" status line of every repeater between the point of damage and the originator of the message. In the case of a damaged circulating token it would trigger every "link error" status line in the ring. A later broadcast request by a maintenance node asking for trouble reports would then produce an avalanche of responses even though only one error has occurred; it may not even be possible to indicate the troublesome link in the case of an error that has circulated all the way around the ring. The choice of replacing a code violation by a correctly formed one-bit is intended to help preserve the integrity of circulating tokens. The notion of link error isolation is borrowed from the Cambridge University ring network [9].)

Level 0.1: Digital Signalling level

1. The data packet is placed on the line using a variant of the token/connector scheme of the V.1.LNI design, with the same "bit stuffing" strategy used to prevent data from appearing to be a level 0.1 signal. The primary change from Version One is the replacement of the packet length field with an end-of-packet signal. The figure below illustrates.



2. An eight-bit sequence known as a flag is a prefix of each level 0.1 signal; the rest of the signal is a sequentially decodable one- or two-bit sequence. The flag consists of a zero followed by seven ones. The bit stuffing mechanism watches all originating data to detect sequences that match the first seven bits of the flag and it inserts a zero bit after the seventh bit of such sequences to insure that the data will not be confused with a flag. The flag is chosen to be no more than 8 bits in length, so that it can be detected in time to serve as an end-of-packet marker, before the early bits of the flag have been transferred to the packet buffer by a copying LNI. This choice avoids two uglier alternatives: exposing level 0.1 signal formats to level 0.2 or else inserting enough buffering between level 0.1 and 0.2 to allow end-of-packet detection before transfer of the front of the flag. (A shorter flag pattern was considered. A shorter flag (say, 3 bits instead of 8) would trigger more frequent bit stuffing with consequent lowering of efficiency of use of the 8.3 bit/sec. signalling rate, but it would also reduce the problem of delay padding for short rings. A three-bit flag would lead to a five-bit token signal, which would "fit" without padding on a ring that has three nodes and 70 meters of cable. The 8-bit flag leads to a 10-bit token signal, which requires delay padding for nets smaller than, say, 5 nodes and 150 meters of cable. To use a 3-bit flag without unacceptable efficiency loss, it might be necessary to introduce circuitry that recognizes that since packets are an integer number of bytes in length, flags must also start at a bit position that is a multiple of 8 bits away from the end of the preceeding connector. Unfortunately, this approach only reduces the padding problem, but does not eliminate it, so the alternative strategy of allowing short rings to operate entirely in contention mode was chosen instead.)

3. In the idle state, the ring continuously circulates a signal known as a token, which consists of a flag followed by the sequence one-one. The part of the ring that is not currently repeating the token signal sees no data.

4. When a node wishes to switch to originate mode, it continues to repeat until it recognizes a token signal, at which point it changes the token signal to a connector signal, switches to originate mode, and appends its own packet, followed by an end-of-packet signal, two signalling bits, and a new token signal. The token signal code sequence is chosen to permit it to be converted to a connector signal by overwriting its last bit with a zero.

5. At the time it switches to the originate mode, the originating LNI stops repeating data, and instead begins to drain data from the ring. It continues to drain data until it has drained its own packet, whose beginning is marked by the connector signal and whose end is marked by the end-of-packet signal. After noting the value of the link parity and refused bits that follow the end-of-packet signal, it switches to repeat mode to pass along the next signal, which started out being the token signal it originated but which may have been changed by now to a connector signal by some other LNI switching to originate mode. Thus an originating LNI destroys a token by converting it to a connector, but then it appends a fresh token to the end of its message for use by others. Note that the end-of-packet signal and the token signal cannot be combined without providing a token signal time's delay (10 bits) at each repeater, because the LNI must switch to repeat mode in time to repeat the first bit of the token signal, but it cannot recognize any signal until it has seen the eighth bit of the flag that distinguishes the signal from data.

6. The "refused" bit is set to zero by the originating LNI, and if level 0.2 of a repeating LNI matches the destination address of the packet but for some reason cannot accept the data, it asks level 0.1 to change the "refused" bit to one. This bit is passed by the originating LNI back to its host, where software can decide how to handle the situation. This feature is provided to allow an originating host to detect that it is overrunning the buffering capability of the copying host. The refused bit will not be set by a repeating LNI if it is refusing a broadcast packet or if it is monitoring all packets. (The V.1.LNI design had two bits, labelled "match" and "accept". The "refused" bit value one corresponds to "match" = 1 and "accept" = 0; a "refused" bit value of zero corresponds to "match" = 0 and "accept" = 0. Having eliminated the multiaddress feature of V.1.LNI, the usefulness of the other two bit combination of that design is not very great.)

7. For trouble isolation purposes, a link parity bit is provided that is recalculated at every repeater. This check bit is calculated and set by the transmit side of each repeater, and checked by the receive side of each repeater; if the link parity check fails the "link error" status line is set for one subclock time. This status is intended to be passed to the host for use in preparing a trouble report message, or other trouble isolation scheme. The link parity bit begins its coverage with the first bit of the packet, and it covers the end-of-packet signal that follows the packet. The value of the parity bit is chosen to create an odd number of one bits. The link parity bit of level 0.1 and the Manchester code violation detection of level 0.0 are somewhat overlapping in function; the level 0.1 link parity bit is provided so that the level

0.0 transmission scheme between any pair of repeaters can be replaced without the need to change the level 0.1 digital signalling format, which would require a change to all repeaters on the ring.

8. There is no hardware checksum provided. This choice was made after considerable discussion, and it may carry some risk. The reasoning for omitting this usually-provided feature involves several steps. First and most important is that it appears that all existing and proposed higher-level protocols will (and must) include a checksum of some kind on their data, to insure correct end-to-end delivery and reassembly of higher-level data structures. This means that a checksum in the level 0 protocol could serve only as a trouble localizer or a performance refinement, perhaps triggering hardware or low level software retry, which can be accomplished sooner and with less fuss than at higher levels of the protocols. Omission of the level 0 checksum means that error detection and possible retry are postponed until higher-level software protocols come into play; the important thing is that in any case all errors are eventually detected. The value of a level-zero checksum is thus primarily as an early-warning performance improver; the frequency of detectable errors is the primary question. In the Xerox PARC Ethernet, checksum failures have been reported to affect one in six million packets [6]. In the Version Two ring, the transmission environment is substantially more benign--it is point-to-point rather than broadcast, and it uses balanced twisted pair rather than unbalanced coaxial cable. It is therefore reasonable to expect error rates at least as low as in the Ethernet, and one concludes that level-zero checksums will not often get a chance to provide their early warning function. (This line of reasoning is corroborated by experience with the Version One ring network--so far, the only observed checksum failures have been traced to failure of the checksum circuitry itself. One reason for this apparently high level of reliability is that if a transmission line or a repeater begins to fail in such a way as to change bit values, it will very soon destroy the continuously circulating token. (Recall that upwards of 70% of the time the token will be the only thing on the ring.) Frequent token losses cause the net to be declared broken and repair is initiated immediately. In effect the circulating token acts as a continuous reliability test, and the checksums on individual packets hardly ever get a chance to fail. Unusually high noise pulses and other one-shot random events are the only mechanisms to produce errors that a checksum is likely to detect before a token does, and such events should be relatively rare in the balanced environment.)

9. Level 0.1 provides a feature to aid in detecting that the control token has been lost. Whenever a token is detected, a watchdog timer is set to 300 msec. If the token is circulating normally, it will pass by more frequently than every 300 msec., and the timer will never complete. Similarly, whenever a digital signalling flag of any kind is detected, a second timer is set to 1.2 msec. If data is circulating normally, this timer will never complete either. The use of these two timers is described later, in the section on the distributed ring control design.

10. The ring initialize feature of the LNI is partly implemented in level 0.1, with the help of host software. The host requests ring

initialization as an option when originating a message. When level 0.1 gets this ring initialize request, it immediately switches from repeat to originate mode, but transmits no data for a time long enough for the next repeater to prepare to resynchronize its receiver. Then it transmits a newly-fabricated connector signal, followed by the data packet supplied by the host and, as usual, an end-of-packet signal and a token signal. This ring initialize mechanism at level 0.1 is only one part of the complete ring initialize mechanism, which is described later.

11. The digital signalling format is designed so that in every case in which data travelling by a repeater should be changed in value, it is possible to determine (and begin transmitting) the new value without knowledge of (and therefore waiting to decode) the old value. This design strategy allows the delay through the repeater to be minimized, which in turn means that a minimum number of components must be "in series" with the ring, a prerequisite to maximum repeater reliability.

12. When the node requests the LNI to reset, the effect at level 0.1 depends on which mode the LNI is in. In repeat mode, the LNI continues repeating. (Note that at level 0.1 there is no difference between copy/repeat and match/repeat modes. All arriving data is passed to level 0.2 for consideration.) In originate mode, the LNI immediately transmits a token signal and switches to repeat mode. It does not transmit an end-of-message signal. This omission insures that the recipient will recognize that the packet may be incomplete.

13. At power-up time, the host holds the reset request line continuously until power is stabilized. Thus at digital signalling level 0.1 starts out in repeat mode.

14. A digital loop-back mode can be entered by setting a status line; when in this mode an originate request causes the digital signalling level to wait until input is enabled, then copy the data coming from the output packet buffer back to the input packet buffer. The purpose of this node is for trouble isolation and diagnosis, so it is implemented in such a way that as much as possible of the digital signalling level hardware is invoked. (This idea borrowed from the Primenet local ring net [10].)

15. When a packet is being copied, its format is checked. Failure of any of the following tests results in the status "Packet out of format" and copying stops:

    a) Flag following connector not on byte boundary.

    b) Flag following connector not an end of message.

    c) Flag following end of message not at proper spacing.

16. Originate mode contains a feature to automatically compensate for a ring that is too short to circulate a token. In originate mode, arrival of the end-of-message signal on the receive side triggers a check to see if the token signal has completely left the transmit side. If not, the ring is closed with a ten-bit shift register rather than a direct connection

from the receive side to the transmit side. The need for artificial delay is reassessed every time a message is originated. The strategy has the advantage that in analog loop back mode one can check out all LNI functions. To avoid the possibility that a long-unused shift register is suddenly cut into the ring only to discover that it isn't really working, the software for each host should always check out the ability for a token to circulate in analog loop-back test mode before joining the network.

(An alternative to introducing a shift register would be to rely on contention-based ring initialization for every packet originated when the ring is too short to hold a token. This alternative has three objections:

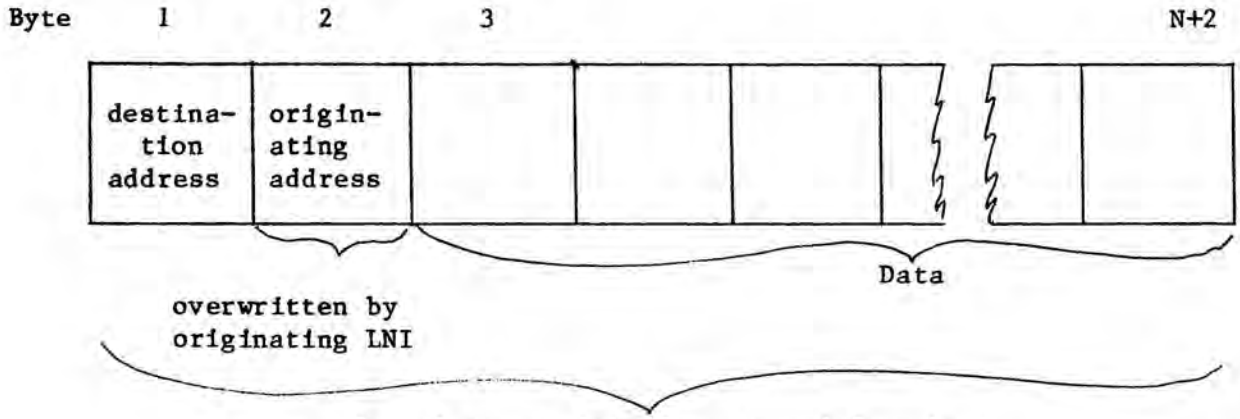
a) The software path that detects the need for ring initialization is forced to be highly-tuned and fast-reacting, because it will be called into operation on every packet originated if the ring happens to be small. If this path were not used so frequently, it could be handled by easier-to-program, but less rapidly responsive, higher level software.

b) The requirement that token loss be detected and recovery procedures be invoked on every packet transmitted over a short ring might have a severe impact on performance. After a packet is sent, the next packet cannot be launched until a 1.2 msec. timeout happens and host software evaluates the situation and reinitiates the transmission with ring initialization.

c) The LNI output machine must be able to drop out of originate mode before it has finished creating the new token signal that goes at the end of its message, in response to a signal from the input side that the front end of the token is arriving. (Otherwise, some other node might catch on to this token, turn it into a connector, and then not be able to recognize it at the front of its own message because its originator didn't repeat the first few bits.) On the other hand, leaving originate mode before completing the token is a bad practice, from the point of view of the originate node strategy of always doing garbage collection and leaving a clean ring behind. If this feature is provided, it could be triggered by noise bits as well as by a short ring, the noise thus triggering more noise.)

Level 0.2: Packet format

1. The detailed format of a data packet is as shown in the figure below. All fields are integer multiples of 8 bits in length, to simplify interfacing to computers that use 8-bit bytes and to take advantage of available logic and memory chips that manipulate 8-bit bytes.



2. The destination address is an 8-bit field, allowing up to 256 nodes to be installed on a single ring. Larger addresses and address match schemes such as used in the Version One design are not used, on the basis that not enough is understood about the real requirements for multiaddress broadcast and automatic internet bridging to warrant placing any particular multiaddress or netwide address scheme in hardware at this time. (Note that the full internet address of the originating and destination hosts is presumably to be found in the packet contents at the next higher protocol level, allowing software implementation of any features that a large hardware address or mask field could provide.) The choice of an 8-bit address, then, is made primarily on the basis of the maximum number of nodes that can be connected on a single ring. The present repeater design has a 1- to 2-bit delay at each node; a ring with 100 to 200 nodes would then have 200 to 400 bit times of delay. This delay is 10% of the transmission time of a maximum-length packet; any more would produce an undesirable performance penalty. (The performance penalty arises because the anti-hogging algorithm forces a once-around-the-ring delay between packets originating at a single node.) A second consideration limiting size of a single ring is that the maintenance and repair of a ring of more than one or two hundred nodes appears unwieldy; a star-shaped ring wire center with more than two hundred cables converging in a single room would require extraordinary management care to keep from becoming a rat's nest.

3. The destination address is supplied as the first byte of the packet by the originating host. (An alternative might be for the hardware to interpret an internet destination address and automatically convert it to a drop address on this ring. This approach would require that an internet address structure be frozen now and placed in hardware, which seems premature considering the range of possible choices that have been suggested for internet addresses.)

4. The originating address is an 8-bit field that is overwritten by the originating LNI with its own ring address. This field is provided to help verify ring initialization and to help clean garbage off the ring. Whenever an originating LNI drains its message from the ring it checks the originating address field to be certain that it is its own address.

The detailed algorithm that verifies initialization and drains garbage is described in a separate section below. There are three other uses for this originating address field: a) it provides a way for a node to discover its own ring address for purposes of informing internet routing gateways. (An alternative would be for the ring address to be stored in the permanent software of the node, which provides too big an opportunity for it not to match the hardware setting. Another possibility would be to provide a special command to read the ring address.) b) The source address field may also be useful in error reporting packets. c) For future experiments with a contention ring, it allows the transmitter to learn quickly that the packet it is receiving is not the same one it is transmitting.

5. The transmitted packet does not carry a length field. Instead, the level 0.1 end-of-packet signal following the packet marks its end. This choice was made for several reasons:

   a) The level 0.1 digital signalling protocol already provides interpacket markers, so a length field is redundant.

   b) A length field would have to be two bytes in extent in order to handle the required maximum packet size. Unfortunately, the byte order of multibyte integers relative to the byte order of character strings is not standardized among different computers, so a multibyte length field would require that the LNI adopt the convention of some computers, to the inconvenience of others. An alternative strategy using an 8-bit field that measures the packet length in 4-byte words was considered and rejected because some scheme for padding out packets not a multiple of four bytes in length would have to be added either to hardware or software.

   c) the digital signalling format is arranged so that an end-of-packet signal is detected before a byte containing the first bit of the signal has been transferred to the packet buffer. (If a larger signal were used, one might object that the lack of a length field with its consequent delay of end-of-packet detection results in exposing the digital signalling level format to higher levels.)

   Perhaps the biggest argument for a length field is that the redundancy it supplies is useful in discovering and isolating ring failures. This argument is less convincing when it is realized that the exhaustion of a down-counter initialized from the length field occurs at a different time than end-of-packet signal detection, so comparison of the two signals require extra effort. Further, there are three other ring trouble indicators in the level 0.1 format (a parity bit, a check for byte alignment of the signals, and a check for proper signal sequence,) and one would expect that most ring failures will be detected by one of these other checks, so a redundant length check should provide at best an improvement of marginal value. (Note that we again presume that the next higher level of protocol has its own end-to-end integrity checks that will infallibly detect any residual errors that slip through the checks of level zero protocol; level zero redundancy has the function of trouble isolation rather than communication integrity.)

6. A destination address of zero is used for broadcast--every node copies all messages that carry address zero. (This limits the number of addressable nodes to 255.) This feature is provided as an experiment--a similar feature has proven useful in the Ethernet as a way of getting started without knowing what part of the net you are attached to.

7. For maintenance and performance analysis, it is possible (by adding a wire) to force an LNI to match every address and thus attempt to copy every packet that goes by. This feature is provided so that a monitoring node might keep statistics of network use and reliability. (Note that unless the receive side of the host interface is double buffered and the monitoring host is very quick on its feet, a monitoring node will not actually be able to keep up with the traffic in a heavily loaded ring network. It is likely that a more highly modified LNI will be required to do complete monitoring. For example, one might modify an LNI to copy only the first few bytes of each packet, and to treat the packet buffer as a circular buffer rather than as a one-packet-at-a-time mailbox.)

8. The address of a node is set by 8 toggle switches, mounted in a single DIP. The multiple name and mask tables of the V.1.LNI are not implemented. The primary reason for this omission is that the usefulness of a name table depends on broadcast--every node must get a chance to inspect the address in every packet to see if it is one of the addresses it is currently supposed to recognize. In a single ring, broadcast comes for free since each packet circulates by every node, but when interconnecting several networks, broadcast appears to require both algorithmic cleverness and extra bandwidth. A second reason for omitting the name table is that in a TTL implementation, it almost triples the number of chips and the board space required. Finally, the arguments for incorporating any particular set of parameters (e.g., name length and number of name table entries per LNI) are not convincing without more experience with applications that make use of the idea. Therefore, it seems premature to freeze any particular design in hardware.

9. Level 0.2 examines the second byte after every connector signal for address match, whether or not the host has enabled copying of messages from the ring. If copying is not enabled, and a destination address match (other than broadcast match or monitoring match) occurs, level 0.2 requests level 0.1 to set the refused bit at the end of that packet.

10. A reset request from the host causes level 0.2 to switch immediately out of originate mode, if it is in that mode.

General considerations

1. The physical packaging of the V.2.LNI is on Digital Equipment style "dual high" boards. Each board has capacity for about 60 16-pin DIP's. The design is modular, in three sections corresponding approximately to the following level 0 protocol sub-levels:

V.2.LNI.X the analog transmission system (level 0.0)

V.2.LNI.CTL repeater and digital signalling controller (level 0.1 and 0.2)

V.2.LNI.HSB packet buffer and interface to host (level 0.H)

V.2.LNI.X and V.2.LNI.CTL is packaged as one board and are expected to use about 45 DIP's; V.2.LNI.HSB is on a second board, with a different implementation for each host computer. Separate documents specify the programming interface of V.2.LNI.HSB for each such implementation. Edge connectors and ribbon cable interconnect the two boards. The first dozen are to be wire-wrapped; it is hoped that later production will be on printed circuit boards, although the dense packing of the 60-DIP boards may make it difficult to accomplish.

2. Power for both boards is derived from the host. All logic operates with (+5,0) supplies. Ground reference is also provided from the host.

3. The V.2.LNI provides a circuit and connectors to energize a relay to allow operation in the star-shaped ring configuration described in NIN 4. (This circuit is called the "I-am-healthy" line.) The default status of this line is de-energized; for it to be energized all of the following must be true:
   - all self-checking circuits of the LNI show no failure
   - the LNI is completely initialized
   - the software has explicitly set on a control bit ("Host ok") that was off at power up time.

   The current flowing in the I-am-healthy line is monitored; when no current is flowing the status bit "node out of ring" is on. When the I-am-healthy line is de-energized, the relay contacts, in addition to bypassing this node, connect the transmit and receive cables of this node together, to allow loop-around testing of the analog transmission circuitry and the cable.

4. The transmission speed is nominally 8.3 Mbits/sec. (This number is one-sixth of the rate of a 50 Mhz clock). This speed is chosen as the maximum consistent with short distances, low noise levels, available low cost logic families, and simple transceiver design. No significant simplification occurs at lower speeds, until one gets below 0.8 Mbits/sec. where some single-chip LSI transceivers operate. Speeds above 10 Mbits/sec. require a more sophisticated receiving strategy (the receiving circuits of the present design would have to sample at rates greater than 60 Mbits/sec., which is pushing the capability of Schottky TTL logic.)

   For comparison with other networks, this speed corresponds to:

   - Raw NRZ bit rate: 16.7 Mbits/sec.
   - Digital signalling rate, using Manchester code: 8.3 Mbits/sec.

- Maximum usable data rate between two hosts considering only level 0 protocol constraints: 5.6 to 8.0 Mbits/sec., depending on number and spacing of repeaters.

The maximum usable data rate is calculated as follows:

	8176	useful bits (maximum length packet)
	314	stuffed bits (assumes data is random)
	16	level 0.2 format bits
	32	level 0.1 format bits
10 to	3500	bit times of round trip delay between packets (5 to 200 nodes)
8548 to	8938	bit times per packet of 8176 useful bits.

Actual file transfer rates will depend on the packaging format of the file transfer protocol, its use of acknowledgements, and the ability of host hardware and software to keep up. The worst-case maximum length of time that might be required for a bit to circulate around the ring from an originator back to itself can be estimated as follows, assuming that each node introduces two bit-times of delay, each node is 150 meters from the wire center (round-trip 300 meters), and there are 250 nodes:

2 bit times @ 125 nsec/bit	250 nsec.
300 meters @ 5 nsec/meter (V = 0.6C)	1500 nsec.
	1750 nsec./repeater

1750 nsec/repeater x 250 repeaters = 0.44 msec.
@ 125 nsec/bit = 3500 bit times

Some 85% of this time is in cable propagation, and one would expect typical installed rings to have much smaller actual round-trip times--perhaps half this amount. The largest possible message contains

8176	useful bits
1024	stuffing bits
48	level 0.1 and 0.2 format bits
9248	bits

and could take as long as 9248 x 125 nsec. = 1.2 msec to transmit, copy, or receive.

5. The V.2.LNI.CTL is designed with a capability of operating at a maximum speed of 16 Mbits/sec. with chip selection. This capability is chosen because it does not seem to complicate the design, and it preserves the option of using the same basic design with a faster (possibly optical) transmission system in the future.

6. Programmable logic arrays are used in the design, and microprocessors are avoided. These two decisions were taken largely to allow an option of future conversion of the design to a single LSI chip. Also, the 8.3

Mbits/sec. data rate (to say nothing of the 16 Mbits/sec. goal of V.2.LNI.CTL) is very hard to meet with available microprocessors.

7. Early in the design of the V.2.LNI it was proposed to generalize its design so that it could be used either as a ring repeater or as the driver of an Ethernet-style transceiver. This proposal was made on the basis that most of the design could be common for the two modes of operation, and only a small increase in complexity would result. The primary reason for this proposal was to allow experimentation with both kinds of networks, Since there is now an Ethernet-like network already in operation at 545 Technology Square, and a genuine Xerox Ethernet to be installed this fall, it is now possible to do side-by-side comparisons of these two approaches without complicating the V.2.LNI design. The proposal has therefore been dropped.

8. As is mentioned occasionally elsewhere in this note, some design considerations have involved leaving open the option of implementing a "contention ring" as proposed in Local Network Note 11. There is no attempt to provide a complete contention ring design at this time, but rather a goal of considering the contention ring requirements enough to avoid accidentally making it impossible to easily convert the V.2.LNI to contention mode operation. (It appears that since ring reinitialization is accomplished by contention, much of the design of a contention ring must be done in any case.) Note that a contention ring does not require a padding strategy since the padding strategy is required only so that the token can circulate. The question of stability of a circulating token also disappears in a contention ring.

9. The LNI is required to be able to copy a packet originated by itself without special case treatment. This feature is to be used for several purposes: a) as part of the ring initialization algorithm; b) as a technique for testing the integrity of the ring. In a large ring, one microprocessor host might do nothing but test the ring once a minute and initiate recovery or repair if the test fails; c) as a way for a node to discover its own ring address at startup time; d) as a hardware debugging aid if data-dependent transmission errors occur; the transmitted and received packets can be compared bit for bit; e) to avoid making self-addressed messages a special case for software to look out for; and f) when the I-am-healthy line is de-energized (and therefore the node bypass relay connects the transmit and receive sides together) for loop-around testing of the transceiver circuitry and cables to the wire center. (This idea borrowed from the Primenet ring network [9].)

10. If a request to originate is pending at the time the LNI leaves originate mode and begins repeating the trailing token or connector signal, the LNI immediately switches back to originate mode if the signal turns out to be a token. This fast turn around feature is provided so that the V.2.LNI.CTL board can be used without modification in a high performance implementation in which V.2.LNI.HSB provides double-buffered input and output paths. Similarly, if V.2.LNI.HSB keeps the input enable request up, V.2.LNI.CTL will copy several immediately adjacent packets if their addresses all match that of this mode. (V.2.LNI.HSB for the PDP-11 and nu-bus do not take advantage of these features.)

Distributed Ring Control: Design

1. Noise, the cutting in and out of bypass relays, and colliding attempts to initialize the ring may produce data patterns on the ring that do not obey protocol. Ring recovery is accomplished by dividing it into two categories, data recovery and control recovery, and using separate strategies for the two categories. Data recovery is concerned with draining accidentally malformed packets and extra digital signals from the ring. Control recovery is concerned with the particular situation when the control token signal gets lost.

2. Data recovery is accomplished by any node that is operating in originate mode; in general, when a node enters originate mode it is prepared to ignore and discard data that does not follow protocol; when it leaves originate mode it attempts to leave the ring in a clean state. Unless another error has occurred while the node was in originate mode, when it exits the ring will either be formatted correctly or else it will be completely drained of data and control signals and the host's software will receive a status report indicating trouble. To accomplish this recovery, originate mode works as follows:

   a) Wait for a token signal to pass by.
   b) Convert the token signal to a connector signal and switch LNI to originate mode. The transmit and receive sides now operate independently and in parallel.
   c) Transmit side: append message, append token signal, then transmit quiet until receive side reaches step i).
   d) Receive side: set "our packet not removed" status on.
   e) Set 1.2 msec. alarm.
   f) Scan and drain input until a connector or token signal is seen.
   g) Check "our packet not removed" status. If still set on, compare source address of packet with this node's address. If they match, set "our packet not removed" to off. Otherwise return to step f).
   h) Scan and drain input until an end of message signal is seen.
   i) Switch to repeat mode.

   If the 1.2 msec. alarm goes off before reaching step h), the LNI switches to repeat mode with "our packet not removed" status still on. At this point it is certain both that the originating packet has been lost and that the ring has been drained. Step f) drains from the ring any noise bits that precede the message train, and step g) removes extra token signals and noise that happens to look like a message train.

3. The need for control recovery is detected during step a), above. If the token signal is lost, the trick is to allow any node to reinitialize the ring and yet not produce an Aloha-like avalanche effect in which 250 nodes are competing to reinitialize the ring, none successfully. For this reason, the initiative to detect and recover from control loss is taken only by a node that is trying to originate a packet, and detection by originating nodes occurs as expeditiously as possible, so that not very many prospective originators are likely to be contending to accomplish recovery. The simplest way to detect control loss is to set a timer before waiting for a token signal. Since every node in a 250-node

ring could happen to be waiting to transmit a maximum length packet (such a packet takes about 1.2 ms to transmit) one could legitimately wait as long as 300 msec. for that token to arrive. However, there are other clues that, if present, indicate much sooner that the token signal has gotten lost. The worst case round trip time for a 250 node ring is 0.44 msec., so if at any point no data is received for this period it is certain that no one is originating data and that the token is lost. The longest message takes 1.2 msec. to transmit, so if data is flowing but a signal flag is not noticed for this period, then the data flowing is not emanating from an originating node, and again it is certain that the token is lost. Finally, if data stops flowing but a token signal was not seen, the token signal has surely been lost. Thus one would expect to be able to detect many cases of control token loss much sooner than by 300 msec. of waiting. The originate mode step a), then, is actually performed as follows:

a1) Check 300 msec. and 1.2 msec. timer. If either has gone off, return the status "ring out of format".

a2) Follow the data being repeated, watching for a token. If a token is detected, proceed to step b)
if either timer goes off while waiting for the token, or a gap follows the data, return the status "ring out of format".

A separate 0.44 msec. timer on lack of data is omitted on the basis that the 1.2 msec. timer will pick up that case almost as fast. Other tests on ring format correctness (e.g., for proper signal sequence or byte alignment of signals) are not relevant here because although their failure indicates trouble of some kind, they do not necessarily indicate control token loss.

4. Ring initialization is accomplished by contention, so as to avoid the need for a central control node. Whenever any node that had planned to originate a packet receives the "ring out of format" status signal, the software at that node takes it upon itself to attempt reinitialization, by simultaneously requesting initialization and origination. The LNI immediately transmits a connector signal and switches to originate mode, starting at step c) of point 2, above. It then continues as normally for originate mode; if some other mode attempts to reinitialize at about the same time both may receive "our packet not removed" status rather than normal completion of the operation; each should wait a random interval and try again. (An alternative strategy of simply forcing a token signal onto the ring and then waiting for it to come back might be simpler to implement, but it appears to lack the immediate verification of correct initialization that the chosen strategy provides.)

Interface to host-specific board

1. The format of the data passed across the interface between V.2.LNI.CTL and V.2.LNI.HSB is the same as the 0.2 packet format. The order of bytes passed across this interface is the same as the order transmitted around the ring.

2. Data is passed between V.2.LNI.CTL and V.2.LNI.HSB in 8-bit parallel form, one byte at a time. Data bit zero is placed on the ring first; V.2.LNI.HSB insures that data bit zero is the least significant bit of the ASCII character representation on the host.

3. The length of the data packet is not passed explicitly; the end of the packet is signalled with a separate status line. This arrangement permits the entire V.2.LNI.CTL to get along without data length counters.

4. Signals to cross between the two boards (34 lines plus ground):

Set by HSB, read by CTL:

HOK	Host OK
LOOPR	Digital loop-back
INITR	Initialize ring
ORIGR	Originate
RESETR	Reset
COPYR	Copy enable
LBO	Last byte out
ODATA0...ODATA7	Data out Bit numbered 0 is ASCII LSB

Set by CTL, read by HSB:

OPRS*	Our packet refused
ROOFS*	Ring out of format
NOORS*	Node out of ring
OPNRS*	Our packet not removed
ORIGC*	Originate complete
COPYC*	Copy complete
NBI	Next byte in
NBO	Next byte out
LBI	Last byte in
POOFS*	Packet out of format
LERR*	Link error
IDATA0...IDATA7	Data in (8 bits) Bit numbered 0 is ASCII LSB

The TTL level chosen to represent signal presence on the lines between the two boards is high for those lines whose names end with an asterisk, and low for the remainder. This level choice, together with the signal senses in the above list, is intended to guarantee that nothing untoward happens if the cable is accidentally disconnected. The intention is that the CTL board will do nothing, and the HSB board will return all possible error status to the host.

5. The order of the bytes transmitted around the ring is the same as the order they are handed to the LNI by the node. (Some local networks require that the node hand bytes to the interface in reverse order, destination last, to simplify buffering. In either approach, the first byte handed to the interface at the transmitting node will be the first byte handed to the host at the receiving node.)

6. The host specific board must drop its originate line at the same time that it sets "last byte out", unless it is prepared to send another packet. If the originate line remains up, V.2.LNI.CTL will attempt to capture the token at the end of the packet just originated, and expect to send another packet. This design is intended to allow V.2.LNI.CTL to be used with double buffered host specific boards for higher performance.

7. The host specific board must drop the copy enable line sometime after the "last byte in" signal comes from V.2.LNI.CTL, and before recognition of the destination address field of the next packet (about 3 byte times,) unless it is prepared to accept two packets in a row. If it is so prepared, it has that interval before the first byte of the next packet arrives.

8. The meaning of the line "Copy complete" is that "Link error" and "Packet out of format" status can be read. The meaning of the line "Originate complete" is that the "Our packet refused" and "Our packet not removed" status lines can be read and that the link parity bit has been checked.

9. The link error line is set for one subclock time whenever a Manchester code violation or link parity error is noticed. The intent is that HSB either set a latch or bump a counter, either of which are readable and resettable by the host.

Host-specific board considerations

1. V.2.LNI.HSB will provide packet buffers of exactly 1024 bytes in length, so as to enforce a uniform maximum packet size restriction, and guarantee that any LNI can receive any packet from any other LNI.

2. The maximum data length is constrained to 1022 bytes, because the 1024 byte packet buffer must hold both the packet data and the two bytes of the level 0.2 protocol. This buffering capacity is the next power of two above the minimum size recommended for TCP internet packets to avoid a requirement to implement reassembly (576 bytes.) It is also large enough to handle a PDP-11, VAX, or nu-terminal 512 byte disk record inside an internet packet. These two considerations, together with availability of suitable memory chips, guided the choice of maximum packet length.

3. The data packet passed to and from the host will be in the level 0.2 packet format, including all fields. it is important to include the destination field in a packet passed to a copying host in order for the host software to learn whether or not this was a broadcast packet and, during trouble diagnosis, to verify that the address matching circuitry is working correctly and that the node address is as expected. The inclusion of the source address field also provides a simple way for a host to learn its own ring address without adding an extra hardware feature for that purpose. (This approach contrasts with an alternative in which the destination and data parts of the 0.2 packet cross separately from the host to the interface and only the data part crosses back; the level 0.2 packet is constructed on the fly by an originating LNI and discarded by a copying LNI; status bits would report special

cases such as "this was a broadcast packet". This alternative approach seems a little ad hoc.)

4. The host-specific board decides, on a host-by-host basis, in which order successive bytes should be arranged before delivery to multibyte memory words. This point should be considered carefully, to make sure that character strings, likely the most common data type, are placed in an order convenient for the host.

Things not yet decided

level 0.0:

1. The need to maintain line balance when extending or contracting bit durations at level 0.0 is questionable. If crystal oscillators are used, one would expect phase corrections to occur in less than 1 bit in 100, and the line charging effect should be negligible.

2. The rest of the digital filter and phase difference estimator of level 0.0 is not yet designed.

3. Choice of a particular transmission cable awaits completion of laboratory experiments.

4. The frequency (and therefore the impact) of metastable states in the level 0.0 line sampling circuit is not yet known. Deglitching circuitry may be required.

5. The V.1.LNI used a ground isolation scheme based on optocouplers driven over the transmission line. This scheme appears not to work at 8.3 Mbits/sec. because of limitations of optocoupler frequency response and the need to supply a large current through the capacitive load of the transmission line. It has been suggested that pulse transformers or isolation capacitors be used instead.

6. To improve the reliability of the bypass relay contacts, it may be useful to run a D.C. bias current through them. This current can be produced by placing drain resistors from each side of the transmission line to the shield, at each receiver, and relying on the transmission coding scheme which supplies a constant average D.C. level to the transmission line.

7. The Manchester code proposed defines a zero to be a downward transition and a one to be an upward transition. There is an alternate code that defines a zero to be no transition and a one to be a transition, in either direction, and yet another in which a one is a transition in the other direction from the last one, while a zero is a transition in the same direction. It has not been evaluated yet whether one of these schemes might lead to a simpler implementation.

8. The LNI bypass relay is yet to be chosen.

other:

9. A programming specification is needed for the nu-bus version of V.2.LNI.HSB.

10. Certain lines of the CTL board should have LED's on them for quick diagnosis of problems. The lines need to be specified. (Power on, Digital loop-back, Data being repeated, Host OK, Node out of ring, token timers.)

Ideas that would be nice to add were it easy

1. A program resettable counter on the number of link error failures.

2. A program resettable counter on the number of packets that this node has refused.

3. For a high-performance host, for which it is important or useful to be able to copy several successive packets in a train all addressed to the same host, rather than complicating the LNI with a double-buffering feature, it may be possible simply to install two LNI's that have the same address. These LNI's can be cross-wired so that the same signal that signifies the end of a message in one LNI triggers the copy enable line of the other LNI, and vice-versa. Some trick would also be needed to coordinate setting of the refused bit.

4. If the V.2.LNI.HSB drops the copy enable line before the last byte has been transferred, V.2.LNI.CTL should set the refused bit. This feature would improve the quality of information carried by the refused bit, but it requires the LNI remember whether or not the reason for copying this packet was exact destination match rather than broadcast or monitor. If an HSB is designed that has less than a full packet buffer, this feature would increase in importance, since it could be used to report to the originator that the HSB buffering capacity was insufficient to keep up with the difference between the ring data transmission rate and the host's data acceptance rate.

5. CTL could pass the node's address to HSB so that HSB could pass it to the host.

Acknowledgement

A majority of the ideas and considerations involved in this design were contributed by Ken Pogran, who also reviewed all of the remaining ideas for compatibility and other system considerations. In-depth analyses of the design, with observations and contributions of considerations that might otherwise have been overlooked, were made by David Reed, Dave Clark, Noel Chiappa, Howard Salwen, and Greg Koss.

References

1. Clark, D.D., Pogran, K.T., and Reed, D.P., "An Introduction to Local Area Networks," Proc. IEEE 66, 11 (November, 1978), pp. 1497-1517.

2. Mockapetris, P.V., et al., "On the Design of Local Network Interfaces," 1977 IFIP Congress Proceedings, pp. 427-430.

3. Metcalfe, R.M., and Boggs, D.R., "Ethernet: Distributed Packet Switching for Local Computer Networks," CACM 19, 7 (July, 1976) pp. 395-404.

4. Saltzer, J.H., and Pogran, K.T., "A Star-Shaped Ring Network with High Maintainability," Local Area Communications Network Symposium, Boston, Mass. May, 1979.

5. Meisner, N.B., et al., "Time Division Digital Bus Techniques Implemented on Coaxial Cable," Proc. Comp. Network Symposium. National Bureau of Standards, Gaithersburg, Md., December 15, 1977, pp. 112-117.

6. Shoch, J.F., and Hupp, J.A., "Performance of an Ethernet Local Network--A Preliminary Report," Local Area Communications Network Symposium, Boston, Mass., May, 1979.

7. McMahon, T.F., "A Contention Ring Implementation of a Local Data Network," S.B. Thesis, Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, January, 1977.

8. Summers, W.I., The National Electric Code Handbook, National Fire Protection Association, Boston, Mass., 1978, p. 807.

9. Wilkes, M.V., and Wheeler, D.J., "The Cambridge Digital Communications Ring," Local Area Communications Network Symposium, Boston, Mass., May, 1979.

10. Gordon, R., (Network Symposium paper on Primenet ring, in preparation).