A PACKET NETWORK ARCHITECTURE FOR LOCAL INTERCONNECTION

Simon S. Lam

Department of Computer Sciences
The University of Texas at Austin
Austin, Texas 78712

ABSTRACT: A packet network architecture for local networking is described. Network nodes are microcomputers connected to coaxial cables, each of which provides a broadcast subnetwork. Subnetworks are interconnected by packet repeaters that serve as switching nodes. A characteristic of the network architecture is the ability to accommodate a wide range of implementation alternatives. Special network features and a new functional architecture of communication protocols are described.

1. INTRODUCTION

The interconnection of computing facilities within a local environment, such as a single building or a cluster of buildings, without the use of common-carrier transmission facilities, is fast becoming an important problem. When the local network distances exceed those of computer system busses (about 100m), the need for packet communication functions such as those used for remote networking [1] arises. Transmission media that have been used for local networking in this context include: twisted wire pair, coaxial cable and fiber optics [2]. The use of a coaxial cable as a passive broadcast channel and contention protocols for multiple access (previously developed for radio and satellite networks [3]) were first demonstrated in Ethernet [4]. Since then, many local networks have been proposed or built with all or part of their transmission facilities based upon broadcast coaxial cables. They include CNET [5], the NBSNET [6], MIT's Laboratory for Computer Science Network [2], FordNet [7] and Cablenet [8], among others.

An experimental packet network for local interconnection based upon broadcast coaxial cables is currently under development at the Department of Computer Sciences of the University of Texas at Austin: the Texas Experimental Network (TENET). The purpose of this paper is to present an overview of the TENET architecture.

The TENET architecture has been defined with the following characteristics in mind: flexibility, reliability, incremental growth capability and upward compatibility.

Flexibility is an important characteristic to us because TENET is primarily intended to be an experimental tool for research on communication protocols. It is thus desirable to have an architecture that can accommodate a wide range of network implementation alternatives suitable for the handling of a variety of traffic environments. Towards this goal, one of our current research efforts is the development of a functional architecture of communication protocols, with rigorously defined interfaces between protocol layers. It will facilitate the experimentation of different implementation algorithms within individual protocol layers to investigate their performance characteristics under different traffic conditions. Basic concepts of the functional architecture are introduced in Section 4.

The types of data traffic that we envision in the local network environments of interest include: interactive computing on time-sharing systems, RJE systems, business systems involving transaction processing and word processing, as well as personal computer systems which use the local network for access to central file systems on mass storage and special processors. (A notable exception here is packetized voice.) With these applications in mind, a key network performance objective is to be capable of handling relatively long data messages for efficient file transfers, in addition to providing a small network delay for short messages and high priority traffic.

Compared to remote networking using common-carrier facilities, a coaxial cable local network has certain characteristics that impact the design of network protocols. First, the transmission speed is significantly higher than common-carrier facilities (500 Kbps using the Zilog SIO chip in a TENET coaxial cable interface; several Mbps can be accomplished with a custom designed interface [4].) Second, the bit error rate of a coaxial cable of about one kilometer in length is considerably lower than that of conventional common-carrier facilities.

The above network design objectives and coaxial cable transmission characteristics are the motivation behind the provision of several special features in the TENET architecture to be described later: (1) a priority mechanism and a pre-emption mechanism in the coaxial cable multiple access protocol; (2) packet repeaters which provide buffering, routing and header switching functions.

2. NETWORK TOPOLOGY

A network implementing the TENET architecture may consist of a single coaxial cable or many coaxial cable subnetworks. Packet repeaters serve as switching nodes that buffer and route packets from one subnetwork to another. The number of subnetworks that can be connected to a single packet repeater is implementation-dependent. The network topology of interconnected coaxial cables is not otherwise restricted. In general, either a tree or mesh topology may be implemented. An example of a mesh topology is shown in Figure 1. Each network node is a microcomputer that interfaces with one or more network users (terminals or computers). The architecture of interconnected broadcast subnetworks has been considered before for several networks [2, 4, 9]. The TENET architecture differs from these networks mainly in a much expanded functional role for the packet repeaters.

Two important functions of (unbuffered) packet repeaters were first discussed by Metcalfe and Boggs [4]. First, since the length of a broadcast cable is limited due to power limitation of transceivers, packet repeaters serve to extend the range of a local network. Second, packet repeaters serve as "filters" that localize traffic within a subnetwork; only traffic intended for another subnetwork will be picked up and retransmitted by a repeater.

The provision for a mesh network topology (with parallel paths between node pairs) in the TENET architecture necessitates the use of packet repeaters which are buffered and are capable of implementing a routing function. Note that since packet repeaters are probably more prone to failure than the passive coaxial cables, it is not clear whether a mesh topology is more reliable than a single cable interconnecting all users. But as we said earlier, the length of a single cable is limited and packet repeaters are necessary for local networks covering an area wider than the maximum length of a single cable. Compared with a tree or dendritic topology (such as those of Ethernet [4] and Cablenet [8])a mesh topology offers more reliability.

When the pattern of traffic among network nodes has a lot of "locality," the filtering function of repeaters enables the network to carry much more traffic than a single broadcast cable. However, those packets that need to go to a remote subnetwork suffer more delay (several hops are required instead of a single hop from source to destination). To alleviate this problem the provision of an optional header switching function in packet repeaters is included in the TENET architecture (see Section 3.2 for description).

Given a local network and a set of network users, there are several ways to partition network nodes into subnetworks. Such partitioning can be done according to the geography of nodes, e.g., all nodes in a single building connect to a single coaxial cable. Alternatively, network nodes may be partitioned according to function, for examples: (1) all nodes belonging to a time-sharing system; (2) all nodes accessing a database system; (3) all work stations of a word processing system. A functional partitioning will probably give rise to a higher degree of locality in the network traffic pattern.

3. SYSTEM ARCHITECTURE

There are three major communication protocol layers in the TENET architecture: data link control, network control, and end-to-end control. The data link control protocol layer is concerned with communication between two nodes or a node and a repeater connected to the same cable. The network control protocol layer is concerned with communication between a source node and a destination node in different subnetworks. The end-to-end control protocol layer is concerned with communication between logical entities called sockets in the source and destination nodes.

Two types of packets are distinguished: (1) intranet packets sent by a source node to a destination node within the same subnet, and (2) internet packets sent by a source node to a destination node in a different subnet. The format of an internet packet is shown in Figure 2. (A more detailed description of the packet format and communication protocol layers will be given later in Section 4.)

We shall next describe data link control functions for individual subnetworks in Section 3.1 and then network control functions involving packet repeaters in Sections 3.2.

3.1 COAXIAL CABLE SUBNETWORK

Each subnetwork consists of a coaxial cable and a population of nodes and repeaters. Each node or repeater connects to the cable via a cable interface (consisting of a tap and a transceiver). For our discussions in this section, we shall view the cable as a broadcast channel shared by a population of users (nodes and repeaters alike). A packet transmitted by a user in the broadcast channel with no interference from others is received by all users and accepted by the intended receiver whose address is indicated in the packet header.

There are two key problems to be handled at the data link control level: multiple access, and scheduling. Suppose there are altogether N users. The multiple access problem is as follows: (a) among the N users, identify those with data who desire access to the channel, the ready users, and (b) assign channel access to exactly one of the ready users if at least one exists. If two or more ready users are present, then the scheduling problem is: which one of the ready users should be given channel access next, in accordance with the system performance objectives.

A difficulty in the design of multiple access and scheduling protocols is that the broadcast channel itself is the only means of communication for coordinating the distributed users. Most coaxial cable networks, including TENET, provide two basic mechanisms, namely carrier sensing and collision detection, for coordinating users in multiple access protocols.

A carrier sensing mechanism enables a user to detect the presence of an ongoing transmission in the broadcast channel. A carrier-sense protocol dictates that a user must not transmit when the channel is sensed to be busy. (The effectiveness of a carrier-sense protocol increases as the maximum channel progagation delay τ between any 2 users decreases. Carrier sensing is effective here because for a cable of, say, 1 Km operating at 500 Kbps, τ is about 2 bit times.)

In TENET, as in most other cable networks, the transceiver in a cable interface can transmit an input signal onto the cable and receive an output signal from the cable simultaneously. Thus when a collision occurs, the collision condition can be detected quickly by each of the transmitters involved; each such transmitter is required to terminate its transmission immediately.

The mechanisms of carrier sensing and collision detection provide the synchronization functions for implementing a variety of multiple access protocols based upon 3 basic multiple access strategies: contention, linear search and tree search. A comparison of these strategies is given in [10] for two extreme traffic conditions: a lightly utilized channel when only 1 out of the N users in the subnet is ready (condition 1); a heavily utilized channel when all N users are ready (condition 2). It is shown that under traffic condition 1, a linear search strategy is much worse than the contention and tree search strategies when N is large. The reverse is true under traffic condition 2. Furthermore, analysis results in [10] show that traffic condition 1 prevails a great deal of the time even when the channel utilization is moderately high, thus favoring contention and tree search strategies when N is large. On the other hand, when N is small, a linear search strategy is desirable because it does not require adaptive control needed for both contention and tree search strategies.

We mentioned that flexibility is a desirable characteristic for the TENET architecture so that it can accommodate a wide range of implementation alternatives for the handling of a variety of traffic environments (or an environment that is not well defined at the time of network construction). For example, the number of users may be large in some subnets and small in some others. A scheduling algorithm more sophisticated than what has been described is desired to handle large packets for applications requiring file transfers as well as to ensure good delay performance for special classes of traffic.

Two additional mechanisms are provided in the TENET architecture for multiple access and scheduling:

(i) A priority mechanism

Packets are differentiated into priority classes numbered 1, 2, ..., P with 1 denoting the highest priority and P denoting the lowest priority. (The TENET architecture allows up to 8 different priority classes.) A <u>deference period</u> of T_p seconds is associated with <u>priority class</u> p, and

$$0 = T_1 \leq T_2 \leq .... \leq T_P$$

(In order for the priority mechanism to effectively discriminate class p and class p+1 packets, it is necessary to have

$$T_{p+1} - T_p > 3\tau$$

where the users are not assumed to be time synchronized.)

At the beginning of an assignment period (which is initiated when a user has just detected the channel changing from a busy to an idle condition and terminated later by a successful transmission) the following algorithm is executed by a ready user with a class p packet to send:

1. Wait for T_p seconds;
2. <u>If</u> the channel has been detected to <u>be</u> busy at any time during the T_p period <u>then</u> do nothing until the next <u>assignment</u> period <u>else</u> transmit immediately;
3. <u>If</u> collision is detected <u>then</u> retry according to a contention <u>protocol</u>.

The contention protocol in step 3 above may be either a slotted protocol (retry with probability p in the next time slot) or an unslotted protocol (retry after a random delay).

Note that the above protocol is a flexible combination of the linear search and contention strategies. The protocol divides ready users with packets to send into different groups. The strategy in general is to perform a linear search over the groups (according to a priority ordering). Once a group has been identified as containing ready users, a contention protocol is used to assign channel access to one of these users.

Consider two special cases. Suppose there are N priority classes with only one user in each class, then the protocol becomes a linear search protocol (according to a prioritized ordering). On the other hand, if only one priority class is implemented, the protocol becomes a pure contention protocol.

The priority assigned to a packet may depend upon many factors. It may be determined at the source who pays for a certain class of service. It may depend upon the packet length, e.g., short packets have higher priority than long ones. It may depend upon the packet type (data versus control).

A priority mechanism similar to the above has been independently proposed and studied by F. Tobagi of Stanford University [11]. It has also come to our attention recently that it is similar to the idea of Acknowledging Ethernet [12] and the protocol of Hyperchannel [13].

(ii) A pre-emption mechanism

The carrier-sense protocol dictates that each user must defer to any transmission currently in progress in the channel. However, in order to accommodate the transmission of very long packets and still provide short delays for short packets and urgent traffic, the TENET architecture provides for a preemption mechanism. There is a pre-emption bit in the header of each packet, which when turned on indicates that the packet may be pre-empted. A ready user with a high priority packet may deliberately generate a collision in the middle of the transmission of a packet with the pre-emption bit turned on. The transmitter of the pre-empted packet aborts its transmission upon detection of the collision and an assignment period is initiated.

The pre-emption mechanism may also be used by a packet repeater to implement a congestion control algorithm. If a packet repeater cannot accept a new input packet addressed to it due to its input buffer limit having been reached (see below), it can use the pre-emption mechanism to stop the packet transmission and then send a high priority control message to the pre-empted user to inform him of its condition.

3.2 PACKET REPEATERS

A packet repeater is connected to two or more broadcast cables. They act as switching nodes and their main functions are the buffering and routing of packets. A significant departure from other packet networks in the TENET architecture is that packet repeaters are not required to do error control at the data link control level (error detection will still be done). There are two reasons for it. First, the bit error rate of a coaxial cable is expected to be quite low [14]. Second, at the relatively high transmission speed of a coaxial cable, buffering at repeaters is expected to be in short supply. This is especially true since one of the network functional objectives is to be able to handle long data packets for applications requiring file transfers.

Header Switching

A consequence of the decision of not doing error control at repeaters, is the new option of performing header switching. As soon as the data link control and network control portions of the header of a packet have been received, a repeater can immediately make a routing decision and retransmit the packet onto another broadcast cable before the packet has been completely received. Header switching when applied under favorable traffic conditions can reduce significantly the source to destination delay of a packet requiring multiple hops; at the same time it will also reduce the buffering capacity required in repeaters. Header switching is similar to the virtual cut-through switching technique analyzed by Kermani and Kleinrock for packet networks using point-to-point links [15].

Routing

The TENET architecture defines a general form of source routing. A variety of adaptive or non-adaptive routing algorithms may be implemented. Each packet contains a route number which is set initially by the source node. At a repeater, the next subnet to forward a packet to is determined by a table look-up procedure, using the destination subnet address and the route number in the packet header. There is also an adaptive routing bit in the packet header, which if turned on by the source, authorizes repeaters to change the route number of the packet if the next outgoing subnet of the original route is congested. The construction of efficient routing table data structures as well as update procedures are research problems currently being investigated by us.

Buffer Management and Congestion Control

Packets may be treated differently by repeaters depending upon two criteria. First, each packet has a priority class specified by its source node, as described earlier. In general, the length of packets with high priority classifications is limited to 256 bytes. The length of packets with low priority classifications can be much longer (up to 2^{16} bytes) depending upon a specific implementation. Note that a low priority packet may be longer than the available buffer space of a repeater and can only be handled by means of the header switching option.

Second, the number of hops (subnetworks) traversed by a packet is indicated in a hop count field in its header. A repeater accepts or rejects an incoming packet depending upon its buffer space availability, and the priority and hop count of the packet. Specifically, the mechanism of input buffer limits [16] may be used for controlling the input of new packets into the network, by making use of the pre-emption mechanism described earlier.

Finally repeaters are also responsible for throwing away very old packets i.e. packets that have a very large hop count.

Comparison With Other Local Networks

Our switching nodes are termed packet repeaters following the terminology of the packet radio network (PRNET) [17]. Packet repeaters in PRNET are different from ours in two respects: they have the error control function but not the header switching option. Ethernet also has packet repeaters which are unbuffered. Ethernet's repeaters have only the address recognition function and act as pure packet filters.

Both Ford Aerospace Corporation [9] and MIT's Laboratory for Computer Science [2] are developing local networks that provide for the interconnection of subnetworks in a mesh topology. Ford Aerospace Corporation is concerned with interconnecting autonomous local networks using gateways and internetworking protocols. The subnetwork concept as discussed by Clark et al. [2] is very similar to ours, except for one architectural difference. In their discussion, a "bridge" is used to connect two subnetworks, whereas in the TENET architecture, each repeater can connect more than two broadcast cables and provide a routing function not unlike that of PRNET packet repeaters and ARPANET IMPS [1].

4. FUNCTIONAL ARCHITECTURE OF PROTOCOLS

Communication protocols implement functions for achieving synchronization of data between a sender and a receiver. In [18] a new model was proposed to structure the set of synchronization functions needed to bridge the gap between what is provided by a communications channel and what is required for communication between remote processes, into a hierarchy of 5 functional levels; each level of functions in the hierarchy relies upon synchronism at the level below achieved over finer grains of time. The 5 functional levels are presented here (from the bottom of the hierarchy upwards).

1) Bit synchronization--consists of functions for enabling a receiver to retrieve a stream of bits from an incoming analog signal.
2) Frame synchronization--consists of functions for a sender and a receiver to delimit the beginning and end of a frame (packet) and to perform error detection.
3) Multiple access synchronization--consists of functions to effect sharing of the set of communications facilities among concurrent "sessions" involving multiple groups of network users.
4) Content synchronization--consists of functions

dealing with the information content of a packet, namely: the encoding/decoding of data and control information, error and sequence control to ensure that a single error-free copy of each packet is delivered from its source to its destination in the proper order.

5) Session synchronization--consists of functions for the establishment and termination of sessions (e.g., dialogues, conference calls, broadcasts etc.), coordination of send-receive status of network users, and carrying out recovery procedures upon detection of nonsynchronous conditions.

Note that bit and frame level functions are needed for the most rudimentary form of communication over a channel. On the other hand, functions at the content and session levels are determined by the kind of service to be provided to network users.

The above hierarchy of functions may be provided in a network by a single protocol layer or multiple layers. (Recall that TENET has 3 layers.) In a multi-layer architecture, each protocol layer may contribute all, part or none of the functions at each level of the hierarchy needed between the two end points of a communications path.

We illustrate the above methodology for structuring functions by showing the functional architecture of a TENET node in Figure 3; a virtual channel service to network users is assumed. The functions at a particular level that are actually required within a protocol layer depends upon: (1) functions at that level provided to it by another protocol layer, (2) functions at that level that it offers to another protocol layer, and (3) functions at that level that are needed internally within itself. In Figure 3, each horizontal arrow is labeled to indicate the extent (none, part, all) of functions at a level provided by one protocol layer to another across an interface between them. Thus, the functions associated with the 5 arrows across an interface characterize the virtual communications channel that is offered to the protocol layer (or network user) on the right hand side of the interface.

In a TENET node, both the bit and frame level functions are provided entirely by its transceiver (in hardware); see Figure 3. Hence, these functional levels are null within the data link control (DLC), network control (NC) and end-to-end control (EC) protocol layers. (In TENET, the bit level implementation uses the base frequency band and double frequency encoding. The frame level implementation uses HDLC flags, bit insertion/deletion and CRC error detection procedures.)

Multiple access (MA) level functions in TENET are provided by all 3 protocol layers. MA functions in the DLC layer are provided by protocols for cable access and scheduling using the mechanisms of carrier sensing, collision detection, priority and preemption. MA functions in the NC layer are provided by protocols for routing, buffering and header switching. MA functions in the EC layer are provided by protocols for (i) multiplexing the logical communication "pipe" between a pair of network nodes into multiple logical channels between pairs of "sockets," for (ii) fixed and/or demand assignment of sockets to network users, and for (iii) maintainence of directories mapping sockets to network users.

We note that both the DLC and NC layers do not provide session and content level functions across an interface to another protocol layer. However, both levels are needed within the DLC and NC layers. In TENET, functions at these levels are implemented as network status initialization and maintainence protocols; two examples are: (1) protocols for the initial connection of a node to a cable (session level functions in the DLC layer); (2) protocols for updating routing tables (session level funciton in the NC layer).

Figure 3 illustrates a virtual channel service provided to network users, with the EC layer supplying all content and session level functions mentioned earlier. A datagram service may also be provided simultaneously with an additional EC layer that contains no session level functions and only limited content level functions (e.g. for encoding/decoding of data and control information but not error and sequence control.)

Packet Format

The format of an internet packet is shown in Figure 2. (The format of an intranet packet is the same but without the fields in octets 7-11 and 13-14 shown in Figure 2). Observe that if we take out octets 4-17 (inclusive), what is left is basically an HDLC format. The address fields in Figure 2 are self-explanatory. Labels of other fields are explained below.

PAD a pad character
FLAG the HDLC unique flag sequence 01111110
NEXT NODE ADDRESS address of receiver on this cable
P pre-emption allowed (P=1), not allowed (P=0)
PRIORITY packet priority level, 000 is highest priority
T packet is internet (T=1) or intranet (T=0)
LENGTH length of packet in octets
H header switching option, yes (H=1), no (H=0)
A adaptive routing option, yes (H=1), no (H=0)
ROUTE route number selected for given destination
HOPS number of subnetworks traversed by packet
CHECKSUM1 checksum for DLC and NC headers
VERSION version number for connection between sockets
SERVICE datagram (00), virtual channel (01)
CONTROL control field, HDLC-like format
CHECKSUM2 CRC checksum for entire packet

5. CONCLUSION

TENET is an experimental packet network currently under development at the Department of Computer Sciences of the University of Texas at Austin. It is primarily intended to be a testbed for trying out research ideas on communication protocols and network control algorithms. A characteristic of the TENET architecture (that we strive for throughout its definition) is the ability to accommodate a wide range of implementation alternatives. Special features of the architecture include: a priority and a pre-emption mechanism for multiple access and scheduling in broadcast subnets, as well as routing and header switching functions in packet repeaters. We have also introduced a new functional architecture for partitioning functions among communication protocol layers and structuring these functions. This methodology is not limited to local networks considered herein but is applicable to network protocols in general.

Acknowledgments

This work was supported by National Science Foundation under Grant No. ENG78-01803. The author thanks Jim Crandell of the University of Texas at Austin for many fruitful discussions.

REFERENCES

[1] McQuillan, J. and V. Cerf, A Practical View of Computer Communication Protocols, IEEE Press, 1978.

[2] Clark, D., K. Pogran and D. Reed, "An Introduction to Local Area Networks," Proc. IEEE, Nov. 1978.

[3] Kleinrock, L., Queueing Systems, Vol. II, Wiley-Interscience, 1976, pp. 360-407.

[4] Metcalfe, R. and D. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," Comm. ACM, July 1976.

[5] West, A. and A. Davison, "CNET —A Cheap Network for Distributed Computing," Dept. of Comp. Sci. and Stat., Queen Mary College, Univ. of London, Report TR 120, March 1978.

[6] Carpenter, R. and J. Sokol, "Serving Users with a Local Area Network," Proc. Local Area Comm. Net. Symp., Boston, May 1979.

[7] Biba, K. and J. Yeh, "FordNet: A Front-End Approach to Local Computer Networks," Proc. Local Area Comm. Net. Symp., Boston, May 1979.

[8] Wood, D., S. Holmgren, and A. Skelton, "A Cable-Bus Protocol Architecture," Proc. 6th Data Comm. Symp. Pacific Grove, Calif., Nov. 1979.

[9] Yeh, J., P. Lehot and G. Steddum, "Hierarchical Design of a Family of Local Computer Networks," Proc. Pacific Telecomm. Conf., Honolulu, Jan. 1980.

[10] Lam, S. S., "A Carrier Sense Multiple Access Protocol for Local Networks," Dept. of Comp. Sci., Univ. of Texas at Austin, Tech. Rep. TR-113, Oct. 1979; to appear in Computer Networks, 1980.

[11] Tobagi, F., private communication, Aug. 1979.

[12] Tokoro, M. and K. Tamaru, "Acknowledging Ethernet," Proc. COMPCON Fall '77, Washington, Sept. 1977.

[13] Thornton, J., G. Christensen and P. Jones "A New Approach to Network Storage Management," Computer Design, Nov. 1975.

[14] Shoch, J. and J. Hupp, "Performance of an Ethernet Local Network, A Preliminary Report," Proc. Local Area Comm. Net. Symp., Boston, May 1979.

[15] Kermani, P. and L. Kleinrock, "Virtual Cut-Through: A New Computer Communication Switching Technique," Computer Networks, Sept. 1979.

[16] Lam, S. S. and M. Reiser, "Congestion Control of Store-and-Forward Networks by Input Buffer Limits -- an Analysis," IEEE Trans. on Comm., Jan. 1979.

[17] Kahn, R., S. Gronemeyer, J. Burchfiel and R. Kunzelman, "Advances in Packet Radio Technology," Proc. IEEE, Nov. 1978.

[18] Lam, S. S., "Data Link Control Protocols," Dept. of Comp. Sci., Univ. of Texas at Austin, Tech. Rep. TR-124, Jan. 1980.

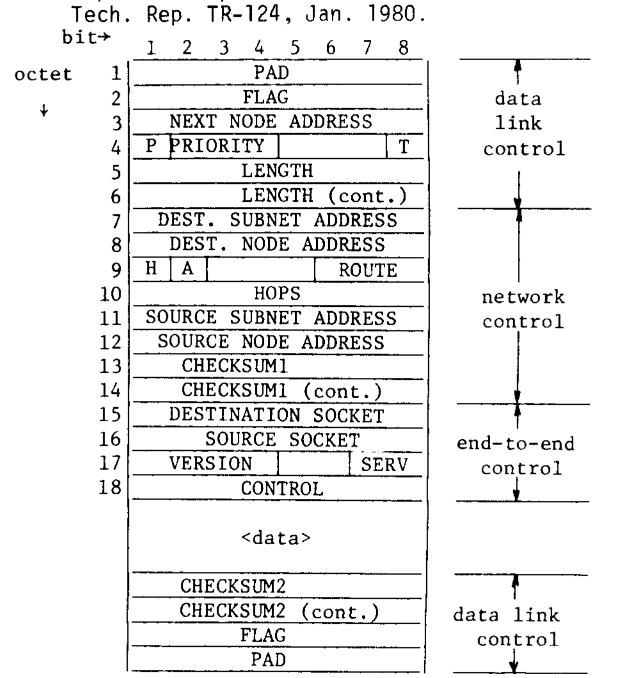

Fig. 2. Format of an internet packet

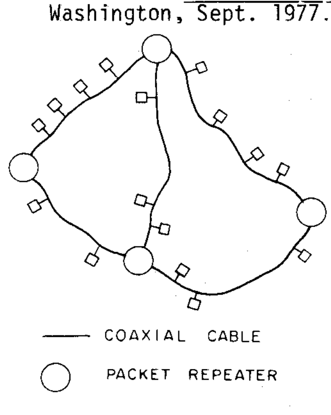

Fig. 1. An example of a mesh network topology.

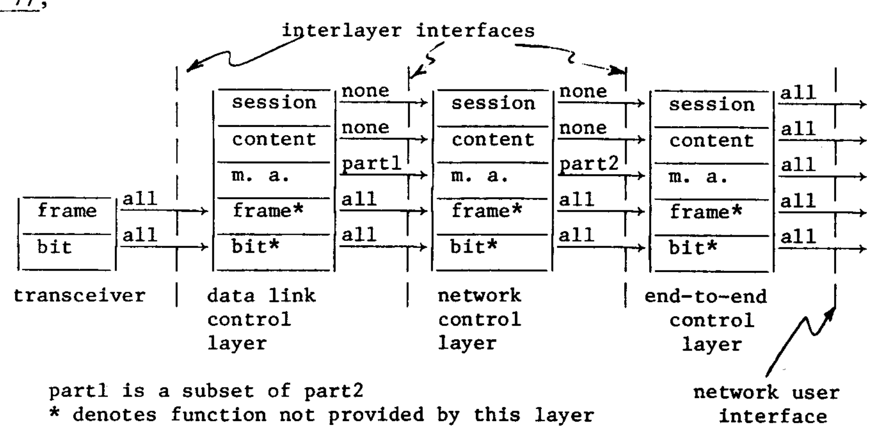

Fig. 3. Functional architecture of a TENET node providing a virtual channel service.