THE BELL SYSTEM TECHNICAL JOURNAL
Vol. 61, No. 7, September 1982
Printed in U.S.A.

Description of Fasnet—A Unidirectional Local-Area Communications Network

By J. O. LIMB and C. FLORES*

(Manuscript received January 15, 1982)

Fasnet is an implicit token-passing, local-area network aimed at supporting high data rates and carrying a wide mix of traffic (data, voice, video, and facsimile). Transmission is unidirectional with stations attaching to the medium passively via directional couplers. A link consists of two lines, one to carry traffic in each direction. Unidirectional transmission provides the potential for efficient operation at high data rates, while the passive medium provides the potential for high reliability. We describe the physical configuration and the protocol and give channel utilization for the condition of continuously queued sources. Mechanisms to control the access of various traffic types are described. Finally, the interconnection of multiple Fasnets is studied for one particular configuration, a ring.

I. INTRODUCTION

Local computer networks operating at 1 to 10 Mb/s are being commercially offered and appear to adequately meet current demands for computer communications within the office environment. However, future needs stimulated by both a broader range of services than is now available and changes in system architecture (e.g., the trend towards distributed processing) could increase significantly the demand for digital capacity. For example, one would like to be able to handle video information, voice traffic, and facsimile, as well as computer traffic, in a single digital system. The availability of a cheap, high-capacity communication conduit between computers will itself stimulate increased traffic. For example, processing time can be traded for communication capacity; rather than transmitting a text file and

* Department of Electrical Engineering and Computer Science, University of California at Berkeley.

formatting it at a remote location, one may choose to transmit a formatted version or even a bit map. Thus, while today 10 Mb/s may be regarded as an extremely generous bit rate for a local computer network, 200 Mb/s may become limiting for an integrated communications network.

Carrier-sense multiple access with collision detection (CSMA/CD) is reliable and reasonably efficient even under heavy load for most conditions.[1] Shoch and Hupp[2] show that measurements of channel utilization of an Ethernet* yield results that are close to calculations made by Metcalf and Boggs, using a simplified model and assuming that active stations have data continuously queued for transmission. Utilization is reported in Ref. 1: $\eta = \gamma/[\gamma + F(M)]$, where γ is the ratio of packet duration to slot time and $F(M)$ is a slowly varying function of M, the number of active stations. Utilization is plotted as a function of γ in Fig. 1, with M as a parameter. If we assume that slot time is 50 μs,[3] and the transmission rate is 100 Mb/s, then for an average packet length of 1000 bits, $\gamma = 0.2$ and utilization is in the range 7 to 8 percent.† Shorter packets, higher transmission rates, or longer slot times would further decrease efficiency. Also note that the above equation does not incorporate source acquisition and synchronization time which, like slot time, is relatively more significant at higher transmission speeds. Thus, it appears that CSMA/CD is not viable for operation at high data rates.[5]

Fasnet is an implicit token-passing protocol developed to efficiently utilize the channel capacity when the ratio of packet duration to the maximum station-to-station propagation time is small (<1). Information flows in only one direction on the medium, unlike the usual CSMA/CD configurations (although see Refs. 5 and 6), but like CSMA/CD the essential passivity of the medium is retained. The access method is closely related to a ring protocol (e.g., see Ref. 7) and may be regarded as a variant of implicit token passing.

Reliability was an important consideration in the design of Fasnet. Consider both the transmission medium and the control electronics. Reliability of the transmission medium may be enhanced by keeping active electronics in the medium to a minimum. Bus architectures such as Ethernet have occasional repeaters, depending on the length of the signal path. Cable-TV (CATV) type architectures have periodic line amplifiers whose spacing is determined by the number of stations (taps), as well as by cable attenuation. Ring architectures usually have

---

* Ethernet is a trademark of Xerox Corporation.

† Since Ethernet requires $\gamma \geq 1.0$ for collisions to be reliably detected, utilization for values of $\gamma < 1.0$ were obtained by multiplying the utilization for $\gamma = 1$ by the value of γ. This can result in efficiencies much lower than that obtainable by other CSMA protocols, e.g., p-persistent CSMA.[4]

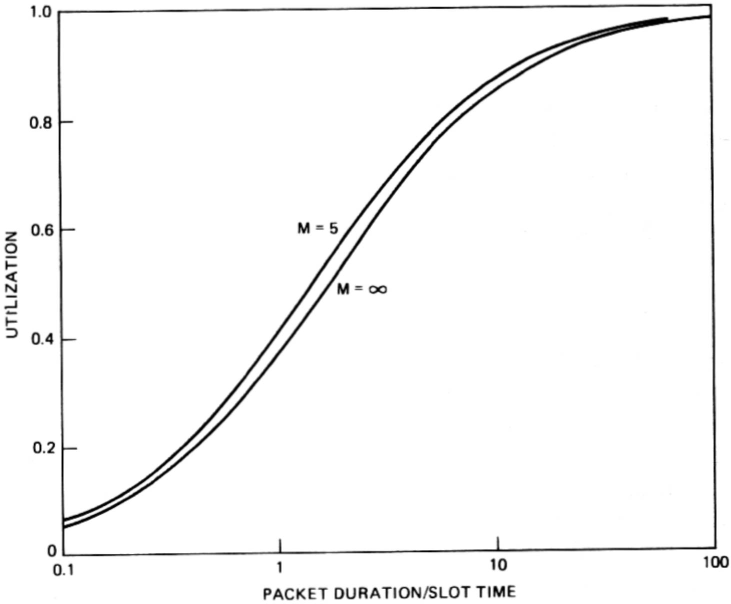

Fig. 1—Plot of efficiency as a function of the ratio of packet duration to slot time. Parameter is the number of simultaneous users. Efficiency changes little as the number of users goes from 5 to ∞.

most electronics in the signal path; digital regeneration is usually provided at each station.[7] Turning to control, reliability considerations tend to favor distributed control. An alternative to a fully distributed system is to permit some stations to perform unique functions but have these functions assumable by any station on the network; however, this can result in a large cost penalty. A further alternative is to have the function performed on a server basis (two or more stations provide the service to all other stations). The Fasnet medium resembles that employed in CATV, and control is primarily distributed with some functions assumable by all stations.

There is considerable attraction in having a single system to handle all forms of traffic in the environment. Indeed, such a system may be regarded as an extension into the local environment of the Integrated Systems Digital Network (ISDN) that is being so vigorously pursued in the long distance and local loop environments by the common carriers.[8] An integrated transmission system simplifies the implementation of services that utilize different types of traffic. Examples are voice-annotated electronic mail[9] and interactive use of voice and facsimile.[10] An integrated transmission system also provides the opportunity to reduce overall transmission needs by taking advantage of the complementary nature of some types of traffic; for example, most electronic mail can be deferred until after the voice busy hour. Further, one would anticipate cost reductions in having one integrated system over a number of separate systems.

While design of an integrated system offers opportunities, it also presents the need for compromises and trade-offs. Consider terminal costs and transmission efficiency. If a system is to provide economical interconnection for telephones and terminals, it must permit construction of an interface that is cost-effective relative to alternative solutions. This may mean that some interfaces have to be tailored to the specific application to make them competitive.

Virtually any type of traffic should be able to exploit the channel efficiently. For example, environments that generate a large number of short messages (e.g., computer terminal traffic), as well as environments that generate a preponderance of long messages (e.g., file transfers), should be able to operate efficiently. This requires that there be a minimum of structure at the lowest common level of the service. For example, a packet structure which mandated a source-address field, while useful for computer traffic, may be unnecessary overhead for a voice channel where call set-up would establish the identity of the source.

The description of Fasnet starts in Section II with the physical loop; the access protocol is described in Section III. The performance of the basic system is given in Section IV, followed by a discussion of some of the system design issues (in particular, the synchronization and signaling procedures) in Section V. Section VI describes variations of the basic system, including methods for improving efficiency, particularly when the number of users is small. Section VII describes mechanisms to support the efficient management and control of mixtures of different traffic types. Section VIII describes the interconnection of Fasnets, with consideration of the impact of the topology on throughput and ability to handle localized sources of traffic. Section IX summarizes the paper.

II. PHYSICAL CONFIGURATION

The basic link, as shown in Fig. 2, consists of two lines. One line passes all stations carrying traffic in one direction and the other line passes all stations carrying traffic in the other direction. For line A, station S_1 is referred to as the head station and S_n the end station. For line B the assignment is reversed. Together the two lines provide a connection between any pair of stations attached to the link. While the lines may be either twisted pair, coaxial cable, or light fibers, we will be primarily concerned with a coaxial cable implementation. Each station makes two connections to each line. A read tap precedes a

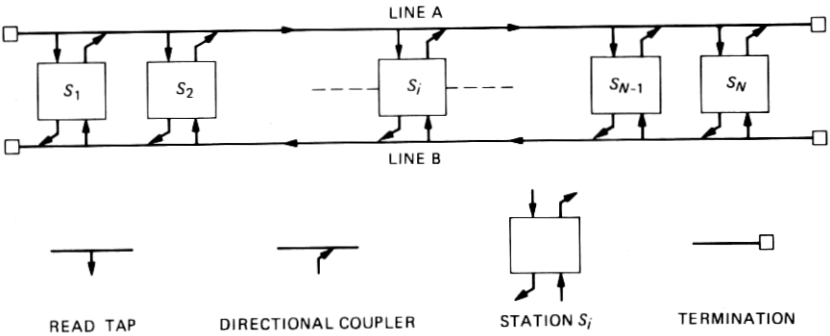

Fig. 2—Physical configuration of a Fasnet link.

passive directional coupler used for writing. The nature of the directional coupler is such that very little energy travels in the reverse direction on the line so that the signal read virtually simultaneously from the read tap will be unaffected by the signal being written on the line via the directional coupler. A station writes on the line by adding energy to the signal already existing on the line. Except for specific fields of the header, the protocol ensures that only one station at a time writes on the line. Thus, once a signal is written on a line, it is not removed or changed by any station. This has certain implications for the line code that is selected (Section 5.2).

Depending on the length of the line, amplifiers are needed to boost and compensate the signal. The technology and design procedures used for CATV systems[11] are directly applicable here, although the noise margin required for a high-quality video signal is somewhat greater than that required for two- or three-level digital transmission.

Links may be joined together to form a network of links. Usually, links will be run in pairs of lines, but this is not always necessary. The advantage of using multiple links is that the traffic-carrying capacity of the network can be increased and reliability may be improved by the use of redundant paths.

An earlier version of Fasnet[12] differs primarily from the system described here in that a link consists of a single unidirectional line that passes each station twice—on the outbound or write side and on the inbound or read side. Each station makes three connections to the line, a read tap for control purposes, and a directional write tap on the write side, and a read tap for recovering data on the read side. The primary advantage of the scheme described here is that the link can carry approximately twice the traffic of the earlier version. A disadvantage is that a station must select the correct line on which to transmit, and this will depend on the relative physical location of the destination.

III. PROTOCOL DESCRIPTION

The data link layer may be divided into two sublayers.[13] One sublayer, the logical link control with which we are less concerned here, provides functions like addressing, windowing, and acknowledgments. The other sublayer, the media access control with which we are more concerned, determines when and how to send information via the physical medium. This is influenced by the media type, the physical configuration, and the technology used.

3.1 Frame format

The frame structure suggested in Ref. 13 and its relation to the data link sublayers is shown in Table I. The information unit is delivered by the network layer. The logical link control appends the source address, the destination address, the link control field for windowing, acknowledgments, and similar functions. We call this unit a packet, and in the work described here we will assume it is of fixed length. The media access control sublayer appends (*i*) the frame check sequence computed on the previous fields for error detection and (*ii*) the access control (AC) field which determines how and when each station may access the physical medium. The main objective in the design of this field is to control access among all active stations in an efficient, reliable, and fair manner. The frame start and frame end delimiters are unnecessary, since the stations are kept in tight bit and frame synchronization (see Section 5.1). The duration of the frame is referred to as a slot.

3.2 Access control

Basic access control for Fasnet is as follows. The head station, S_1, initiates a cycle on line A. After a cycle has been initiated, each active station on the line with packets destined in the right direction is allowed to access the line for one slot. To do this, each station monitors the line. When it senses the line idle, it seizes the line for one slot. It has to wait for a new cycle to be initiated before it attempts to access the line again. The exact manner in which this is done efficiently and

Table I—Protocol and frame structures

	Protocol Structure	Frame Structure
Data link layer	Logical link control sublayer Media access control sublayer	DA/SA/LC/IU FS/AC/DA/SA/LC/IU/FCS/FE
Physical layer	Physical layer signaling	FS/AC/DA/SA/LC/IU/FCS/FE

FS:	Frame starting delimiter	LC:	Link control field
AC:	Access control field	IU:	Information unit from network layer
DA:	Destination address	FCS:	Frame check sequence
SA:	Source address	FE:	Frame ending delimiter

fairly is described in the next paragraphs. If a station has priority, it is given permission to access the line for an integral number of slots. In this manner, the active stations can access the line for a specified duration in the order in which they are physically located on the line. The operation on line B is identical to that described above with S_N replacing S_1 as head station.

To describe the operation in more detail, let $\{S_1, S_2, \cdots, S_N\}$ be the set of stations in the order of their physical locations as shown in Fig. 2. Let AQ_i and BQ_i be the number of packets queued at station S_i for access to lines A and B, respectively.

When the next packet arrives at S_i from the Network layer interface,
if destination address $j > i$, then AQ_i is incremented by 1;
if destination address $j < i$, then BQ_i is incremented by 1.

The structure of the AC field is shown in Fig. 3. Let t_{fn} be the start of the nth frame. The AC field is from t_{fn} to t_{bn}. Station S_i gains access to line A in the following manner. Let S_i be permitted access for p_{max} packets each cycle. At t_{fn}, the start of the nth frame, the read tap reads the START bit of the AC field. The start of cycle is indicated by START = 1. Because of gate delays in the decision circuitry and propagation delays in the tap cables, the outcome of the read operation is only known at t_{sn}. This additional time of duration τ_{dec}, shown in Fig. 3, is of the order of a few bit times for a 100 Mb/s line and nanosecond logic. Next, the station may simultaneously read the BUSY bit via the read tap and write BUSY = 1 via the directional coupler. Again, the outcome of the read operation is only known at t_{bn} after a delay of τ_{dec}. Nonetheless, the write operation does not alter the BUSY bit if it is already set to 1. The nature of the signaling to achieve this is explained in Section 5.2. At t_{bn}, if BUSY = 1, the station defers until the BUSY bit of the next frame. If BUSY = 0, the station accesses the line for the remaining frame duration.

Station S_i is said to be in one of four states:

IDLE —if it has no packets to transmit, i.e., $AQ_i = 0$.

WAIT—if it is waiting for the start of cycle.

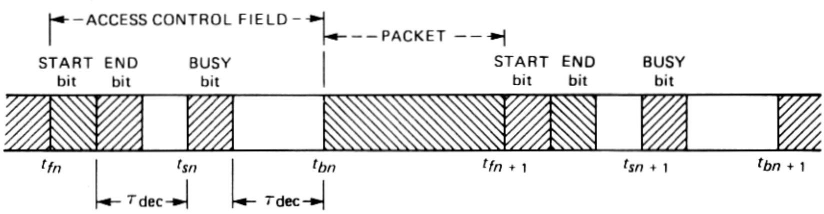

Fig. 3—The frame structure of Fasnet. Each frame consists of (i) an access control field containing START, END, and BUSY bits; and (ii) the packet as provided by the logical link sublayer.

DEFER —if it is deferring to busy users who are upstream on the line.

ACCESS—if it is accessing the line.

The station makes transitions (denoted as →) between states as follows (Fig. 4): While $AQ_i = 0$, S_i is in IDLE. Upon arrival of a packet for line A, $AQ_i > 0$ and $S_i \rightarrow$ WAIT. The station reads the START bit of every frame. When START = 1 $S_i \rightarrow$ DEFER, and the station simultaneously reads and writes the BUSY bit as described above for every frame. When BUSY = 0 $S_i \rightarrow$ ACCESS. Now it accesses the line for p_{max} frames and also writes BUSY = 1 for each. Then $S_i \rightarrow$ WAIT. The station may cease to access the line earlier if $AQ_i = 0$, whereby $S_i \rightarrow$ IDLE.

Station S_1 initiates cycles by START = 1 in the first frame of each cycle. There is an additional bit, END, in each frame to indicate the end of cycles. This bit can be conveniently located in the blank portion of the frame after the START or BUSY bits. When station S_N reads BUSY = 0 on line A (indicating that all active stations have accessed the line), it sets END = 1 in the next frame on line B. On receipt of this frame on line B, S_1 then initiates a new cycle on line A. Thus, in the worst case, line A will be silent once every cycle for a time equal

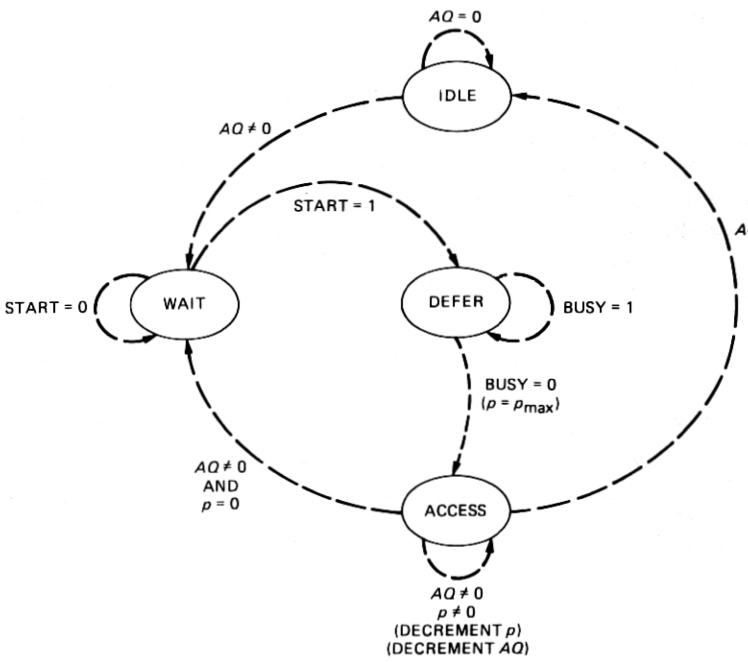

Fig. 4—State transition diagram describing the operation of a Fasnet station.

to twice the end-end propagation delay, plus twice the frame duration, as each end station has to wait until the next frame to set the START or END bits.

The operation on line B is identical, with the roles of S_1 and S_N reversed. Thus, the two lines cycle independently of each other with access being passed between the stations in the same order as their physical locations on each line.

In the protocol described above, the outcome of the read operation on the START bit needs to be known before the BUSY bit is written so that the first frame of a new cycle does not remain idle. Should the START and BUSY bits be adjacent to each other, a station will only learn that START = 1 after the BUSY bit has passed and the frame will not be used. However, for large cycle lengths and short packet lengths, the reduction of one decision interval, τ_{dec}, per frame would be greater than the addition of the extra idle frame.

A further alternative is to have the first frame of each cycle contain only an access field. However, unequal frame sizes complicate synchronization for a very small increase in efficiency.

3.3 Error recovery

The protocol is controlled by the START, BUSY, and END fields. An error in a BUSY field will have no lasting effect; it will result in a packet being overwritten if the busy bit is changed from a 1 to a 0. Alternatively, an empty slot will go unused if the busy bit is changed from 0 to 1. Of more significance is an error in the START and END fields. If a START field is set to 1 in error, two STARTs or a START and an END would be simultaneously present on the loop.

It will be shown that generation of additional STARTs and ENDs will not propagate and have little effect on the operation of the link. We will assume that end stations do not generate STARTs or ENDs that are closer together than the round-trip delay time, τ_r; under normal operation this cannot occur. A false START = 1 will occur either in the active portion of a cycle (including the first empty slot) or in the empty slots occurring at the end of the cycle. If the former, a new cycle will start before, or as, the old one is finishing. Since the additional START = 1 will not generate an END = 1 on the return line (because no transition from busy slot to empty slot is detected), the condition will not propagate. If the false START = 1 occurs in the empty slots, other than the first, the new cycle will start prematurely (actually increasing utilization temporarily). One of two conditions results:

(*i*) The busy part of the additional cycle terminates at least one slot before the next normally occurring START = 1, in which case the end station will detect an end condition. However, because the period since the last END = 1 is less than τ_r a new END = 1 will not be generated.

(*ii*) There is no empty slot before the next normally occurring START = 1. As a result, an additional end condition is not detected by the end station.

Thus, a START = 1 resulting from a fault condition will not produce additional END = 1 bits on the return line. On the other hand, END = 1 faults, unless they are closer together than τ_r, will produce additional START = 1 slots which as just described, have a transient effect on the operation of the link.

Consider the condition where a START or END bit is changed from a 1 to a 0. A new cycle would fail to initiate. After a time-out greater than the longest permitted cycle time, the head station will issue a START = 1, and the link will continue to operate normally. Should a head or end station fail, the station next to the head or end station would assume the functions on detecting loss of timing or after timing out on the arrival of START = 1 or END = 1.

3.4 Fault diagnosis

The independent lines of the Fasnet link provide the opportunity to localize and mitigate some types of faults. Consider first that a line is severed because of some catastrophic event or something trivial like a cable connector failing. The result will usually be either a short or open circuit leading to a gross impedance mismatch. The fault will most likely terminate all effective communication on the upstream side of the fault because of reflections from the mismatch traveling back into station interface units via the read tap. The downstream segment will be affected very little because the directional couplers will propagate little energy in the direction of the mismatch. Thus, a diagnostic program in the end station can determine between which stations the mismatch lies. This is done by having the end station send a query to each station via the intact line and determining which stations respond to the query.

A difficult type of station fault is to have a station continuously write garbage on a line. Diagnostic programs in the end stations, again by querying each station, can determine the faulty station and remove it from service. The head station on the line with the fault, after being informed of the fault by the end station, via the other line, queries each station in turn. If the station fault is confined only to the write circuit, the faulty station will respond. The next station on the downstream side will not respond, since it will not be able to read the query sent by the head station because of the interference from the faulty station. If both read and write circuits in the faulty station are affected, the last correctly responding station will be the station on the upstream side of the faulty station. Thus, the fault is isolated to one of two stations. Both stations may then be disconnected by means of a control signal sent via the functioning line. The faulty station may then be uniquely determined by returning one of the stations to service. If the fault condition resumes, the returned station is faulty and is disconnected; otherwise the other station is faulty.

IV. PERFORMANCE

4.1 Sample operation

Typical operation of Fasnet for lines of 2.5-km individual length, 100 Mb/s bandwidth, and 200-bit frame length is shown in Fig. 5. It shows the time-space relation of the frames on each line. The horizontal axis represents time divided into slots $A_1, A_2, A_3, \cdots$ for line A and $B_1, B_2, B_3, \cdots$ for line B. The vertical axis represents the physical locations of the active stations S_1, S_2, S_3, S_4, and S_5 with S_1 and S_5 serving, additionally, as end stations. The electrical line length is five frames. Station S_1 initiates the cycle in frame A_1, and access passes from S_1 to S_2 to S_3 to S_4. When the end station, S_5, senses BUSY = 0 in frame A_5, it sets END = 1 in frame B_9. Receipt of this frame by S_1 causes it to

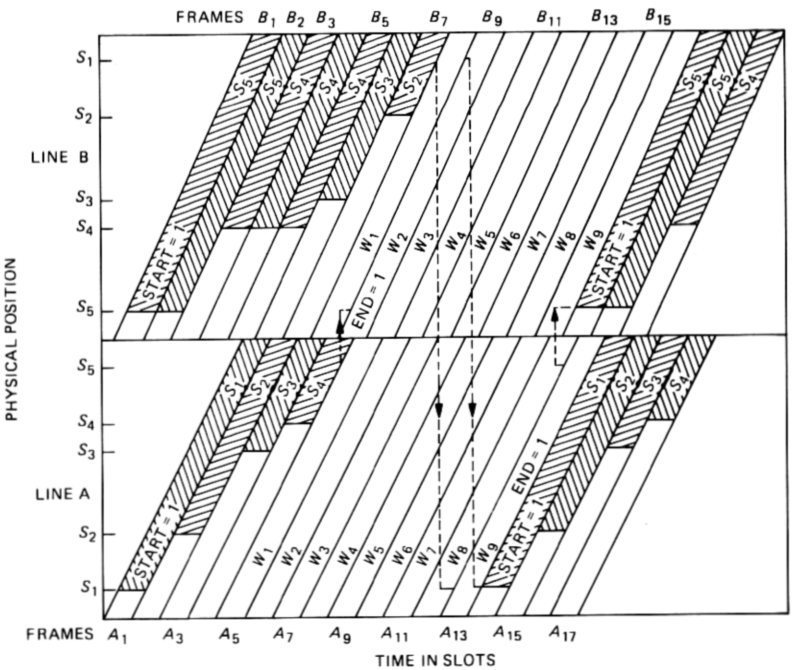

Fig. 5—A graph of activity on a Fasnet link (lines A and B) as a function of time. The dotted lines indicate the flow of information from one line to another.

initiate a new cycle in A_{14}. Similarly, a cycle on line B starts at B_1. Assume that S_5 and S_4 are permitted access for up to two and three packets, respectively. Station S_1 senses BUSY = 0 in frame B_8 and sets END = 1 in A_{13}. Receipt of this frame by S_5 causes it to start a new cycle in B_{17}.

4.2 Utilization

As there are no collisions, no capacity is lost through collision resolution. However, the utilization is not 100 percent as each line is idle at the end of each cycle. The idle period is nine frames, W_1, W_2, $\cdots$ W_9, on each line in Fig. 4. In the worst case, this is equal to twice the end-end propagation delay, plus twice the slot time (one slot time on average) as each end station has to wait until the start of the next slot to set the START or END bits. If

ν = speed of propagation on the line (m/s)

W = line capacity (b/s)

L = line length (m)

F = frame size (bits)

M = number of busy stations with downstream traffic,

then if each station is allowed access for only a single packet per cycle,

$$\text{cycle length} = \tau_c = M^*(F/W) + 2^*(L/\nu) + (F/W)\,,$$

$$\text{duration of busy frames} = \tau_b = M^*(F/W)\,, \tag{1}$$

and

$$\text{utilization} = \eta = \frac{M^*(F/W)}{M^*(F/W) + 2^*(L/\nu) + (F/W)}. \tag{2}$$

The effective utilization is lower since a fraction of each frame is devoted to access control. However, for large F, it results in only a small reduction in utilization. If $\nu = 2.5 \times 10^8$ m/s, $W = 100 \times 10^6$ b/s, $L = 2.5 \times 10^3$ m,

$$\eta = \frac{M^*F}{(M+1)^*F + 2000}.$$

For $M = 100$,

$F = 50$ $\quad\eta = 71$ percent

$F = 100$ $\quad\eta = 83$ percent

$F = 200$ $\quad\eta = 90$ percent

$F = 500$ $\quad\eta = 95$ percent

$F = 1000$ $\quad\eta = 97$ percent.

For the same values of ν, W, and L, with $F = 500$[1,2] and assuming an Ethernet slot time $T = 50 \times 10^{-6}$ s,[3] we can compare the performance of Fasnet and Ethernet as the number of stations is varied as shown in Table II.

Unlike Ethernet, Fasnet has the desirable feature that as the load increases, the utilization also increases. The above figures do not reflect the fact that in practice the length of packets is variable and, consequently, the fixed frames of Fasnet frequently will be only partially filled. The effect on η depends on the distribution of packet size, and to some extent is determined by the system design. For example, in a system designed for large amounts of voice traffic, F could be set equal to the size of a voice packet.

V. IMPLEMENTATION CONSIDERATIONS

The design criteria previously stressed in the introduction affect the implementation in important ways. In particular, the requirement to operate at high speeds and the unidirectional operation of the bus affect the design of the synchronization system; in turn, the type of synchronization and the use of directional couplers impact the choice of the line code that is used.

5.1 Synchronization

Bus systems in which signals travel in both directions on the line require the receiving stations to adapt to the signals transmitted by the sending station because the amplitude, dispersion, and phasing of the received signal vary depending upon the position of the transmitting station on the line. Synchronization can be achieved very quickly when the signaling rate is low relative to the bandwidth of the transmission medium. At higher signaling rates, synchronization needs to

Table II—Fasnet versus Ethernet as a function of number of busy stations

M	Fasnet (in percent)	Ethernet* (in percent)
5	50	4.1
10	67	3.9
50	91	3.7
100	95	3.7

* Note that since the minimum permissible packet length is 5000 bits, η is calculated as 0.1 of η with 5000-bit packets. Other CSMA protocols that do not require collision detection perform better.

be more accurate to achieve good error performance. Ethernet specifies a synchronization preamble of 64 bits and for higher transmission rates an even longer sequence may be required. Thus, for short messages efficiency would be significantly reduced. Using a unidirectional bus, each station can be synchronized to a common clock issued from the head station. Thus, if all stations add signals to the cable in phase with the transmitting clock, stations will receive the signals in correct phase. Similarly, fixed gain and frequency compensation can be employed. The problem of reliability can be overcome by giving each station the ability to supply clock. The clock drive would be inhibited by detection of, and locking to, an incoming clock.

Initial tests have shown that a simple, cost-effective method of synchronization is to synchronize to a continuously injected pilot tone placed at the high end of the signaling band. The synchronizing function then assumes a negligible fraction of the transmission capacity.

In addition to bit synchronization, frame synchronization is also required. This is achieved by sending periodically a synchronizing bit pattern. Design is simplified if this is sent after an integral number of frames, say 64 or 128. With tight bit and frame synchronization, successive frames may be butted together without a gap.

5.2 Signaling

Because synchronization is achieved independently of the data signal, line codes with fewer transitions may be considered. It is particularly convenient if a code is chosen that couples no energy to the line when one of the logic states is continuously transmitted (assume logic 0). Each station at the end of transmission then simply returns to logic 0, and there is no need to "disconnect" the transmitter from the line. The two line codes we are investigating are a bipolar three-level code, Fig. 6a, and a nonreturn to zero (NRZ) two-level code, Fig. 6b. The two-level signal has a greater noise margin; however, one has to contend with the dc signal component.[14]

The Fasnet protocol does not permit subsequent stations to modify a signal already transmitted by an upstream source. In principle, this could be done. For example, a signal from one station could be deleted from the line by a second station writing the complement on the line. In practice, signal levels would have to be matched very accurately for such a scheme to work.

There is one condition in which more than one station may add energy to the same bit in a frame—the BUSY bit of the AC field. As a result, the amplitude of this bit may far exceed the amplitude of the remaining signal. This may lead to errors in adjacent bits of the AC field. To prevent this, guard bands on either side of the BUSY bit

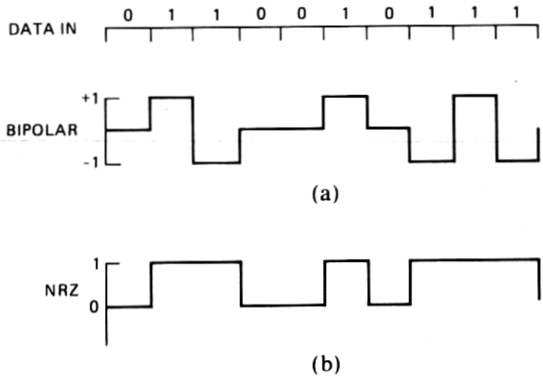

Fig. 6—(a) Bipolar three-level line code. (b) NRZ two-level line code.

should be used. Notice from Fig. 3 that the access field is configured so that the guard bands fall in the intervals τ_{dec} and, in practice, will have a comparable duration.

VI. IMPROVING UTILIZATION

As can be seen from (1), efficiency increases (*i*) as cycle length increases and (*ii*) as the idle period at the end of each cycle (intercycle gap) decreases. At the expense of some increase in complexity, techniques may be devised to improve utilization by increasing cycle length or reducing intercycle gap.

6.1 Control of cycle length

Since START = 1 may be read by all stations, the length of the last cycle, τ_c, may be determined by any station. As previously described, each station may transmit up to p_{max} packets per access. Thus, by controlling p_{max}, stations may influence the value of τ_c. Station control of τ_c by manipulation of p_{max} is obviously limited. For example, let us assume that p is fixed at 1 and that we have stations each generating packets at a rate $< 1/\tau_c$. Increasing p will not change the cycle length since packets will be transmitted before a queue can form. On the other hand, increasing p for heavily loaded stations will lead to an increase of τ_c, provided τ_c is less than the accepted maximum.

6.2 Reducing inter-cycle gap

Three methods are described for reducing the intercycle gap and hence increasing the line utilization. In the first, stations detecting the END = 1 bit seize empty slots on the other line; in the second, stations use the END field as a request field; in the third, the end station

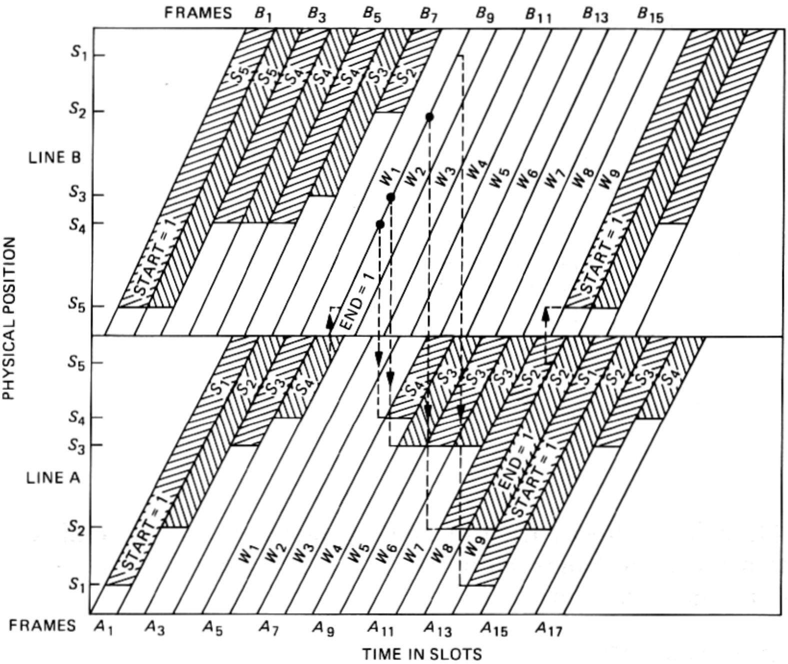

Fig. 7—A graph of activity on a Fasnet link as a function of time. Empty slots encountered on line A by stations that have read END = 1 on line B are utilized. The IDLE period is reduced to three slots. Corresponding use of IDLE slots on line B would also occur but is not shown.

attempts to estimate the end of a cycle, setting END = 1 before BUSY = 0 is received.

Considering the first method, any station S_i in the WAIT state that observes END = 1 may attempt to seize any empty slots on the opposite line.[12] The number of empty slots seized depends on the time the END = 1 frame takes to propagate to the next active station, which then seizes empty slots, thus preempting active stations downstream.* The intercycle gap now depends on the propagation time from the last active station to the end station and back. (The relative timing of the frame starts in the two lines will also affect the gap size). As shown in the example of Fig. 7, the intercycle gap has been reduced from nine slots to three, shown for line A only.

In the second method, stations in DEFER and ACCESS states write REQUEST (REQ) = 1 on the return line (the END field is now

* A station may only transmit a single frame at a time because of the possibility of preemption. This will interfere with construction of "superpackets." Superpackets are used to increase efficiency by reducing the effect of packet overhead. The overhead is attached only to the first packet of the superpacket.[12]

replaced by an REQ field). After all stations have been served, the head station will read REQ = 0 and initiate a cycle. To ensure that at least one REQ = 1 occurs in a cycle for which only one station is active, a station in changing from WAIT to DEFER or ACCESS writes at least one REQ = 1. Approximately speaking, the average intercycle gap is now equal to twice the propagation time from the head station to the last active station, plus half a slot time. Notice that this procedure is more distributed in that the end-station function of recognizing the end-of-cycle END condition and writing END = 1 on the return line is now bypassed. Each station now performs an equivalent operation.

A further refinement is to observe that the head-station function can also be distributed.* Each station with traffic to transmit need not wait for the head station to issue START = 1. Rather, after reading REQ = 0 on the return line, it can switch from WAIT to DEFER or ACCESS mode setting the p register to the allowed number of packets that may be transmitted per cycle. The intercycle gap is virtually eliminated under heavy traffic conditions and for a large number of users as in the first method. For two continuously queued stations, the intercycle gap is twice the delay time between the stations, plus one slot time on average. Notice that even though no START is being issued, or is necessary, the loop cycles. However, as in the first method, packets from a source during one cycle will usually not be consecutive. The station protocol is summarized in Fig. 8 by means of the state diagram. As seen in comparison with Fig. 4, the protocol is more complex; however, the algorithm no longer requires the centralized head or end-station functions. The issue of distributed control versus centralized control is particularly important in efficiently accommodating different types of traffic and is discussed further in Section VII.

In the third method, each active station in the DEFER state indicates its desire to transmit by setting an additional request field to 1 (REQ = 1) in the access field of the line on which it wishes to write, as in Ref. 12. The end station estimates the cycle length and transmits END = 1 timed to arrive at the head station as the last slot to be used in the cycle is leaving the head station. If the estimate was too low, the end station will read REQ = 1 in the last frame in the cycle indicating that the estimate of the length of the last cycle was too short. Therefore, the estimate of the length of the next cycle would be increased. If the estimate is too high, there will be empty slots prior to the arrival of the next START = 1, and the estimate of the length of the next cycle would be decreased. If the estimate is correct, then the

* Suggested by Z. L. Budrikis in connection with the earlier single-line version of Fasnet.

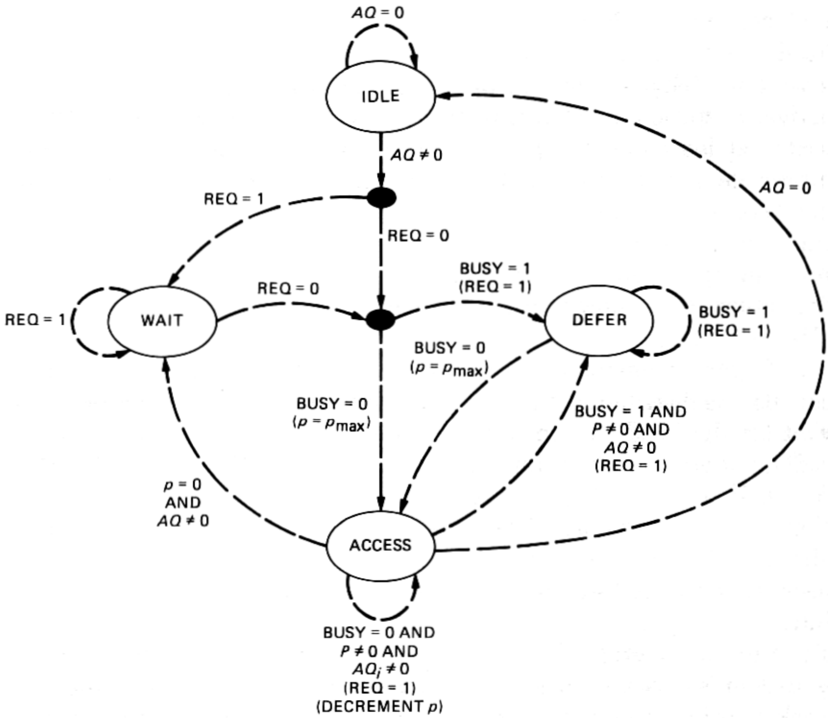

Fig. 8—State transition diagram of a distributed version of the Fasnet protocol in which unique functions in the head and end stations are not employed. Statements in parentheses are actions executed on making the transition. The filled circles denote intermediate states.

last frame before the next START = 1 will have REQ = 0 and the estimate of cycle length would remain unchanged.

Of the three methods for reducing intercycle gap, the former two are distributed while the latter requires additional intelligence in the end station.

VII. TRAFFIC CONTROL

Different types of traffic have very different transmission requirements. Voice traffic is an interesting example of where an initial restriction of access (blocking) is preferable to losing voice packets. Real-time traffic such as voice and video requires a guaranteed maximum delay if information is not to be lost. In contrast, some types of computer traffic are rather tolerant of delay. It is not our aim to develop here a comprehensive algorithm for handling different traffic types; it would depend closely on the particular environment and the mix of traffic. However, we would like to indicate the mechanisms that Fasnet can support to control integration of traffic.

We can identify four different types of control mechanisms.

(*i*) Selection of traffic—The ability to individually control a class of traffic.

(*ii*) Request for access—The ability to communicate that service is required.

(*iii*) Blocking of traffic—The ability to prevent traffic of a specific class from gaining access to the system.

(*iv*) Continuation of service—The ability to delay traffic for later transmission. We will take these four control mechanisms in turn and consider how they may be implemented in Fasnet.

(*i*) Selection—We can borrow from the strategy used by Frata et al.[15] and Ulug et al.[6] START, instead of being a single bit, can be expanded to a multibit word. A START code can then be allocated to each class of traffic. A station would then be permitted to transmit only if the START corresponded to the class of traffic it is waiting to transmit. A class of traffic could denote a traffic type, as well as a priority. The term subcycle will be used to denote the period from a START of one class to the next occurring START. Cycle will be reserved for the period between two STARTs of the same class.

(*ii*) Request—The amount of request information can vary from knowing exactly which station wants to send what traffic in one extreme to knowing just that a station somewhere wants to send some type of traffic in the other extreme. A compromise would be to provide a request word adjacent to the END field in each packet on the return line. Each bit in the request word would denote a class of traffic. This information would enable the head station to determine that one or more stations required service of a particular type without indicating the extent of the demand. Information about demand for service would most likely be used to adapt the control strategy in the case where there was a change in the balance between traffic types.

(*iii*) Blocking—The channel capacity allocated to a class can be controlled by the time allocated to that class as suggested in Frata et al.[15] By issuing a different START after a given period, further traffic of the original class type wishing to transmit would be denied access. Blocking is typically used to handle overflow of the type of traffic that generates information periodically, such as real-time voice and video, and synchronous data traffic. In this instance, it is appropriate to speak of connections between source and destination which implies guaranteed access once a connection has been allocated.

To discuss the allocation of connections in a blocking traffic class requires that the previous definitions of WAIT state and ACCESS state be generalized:

WAIT—A station is waiting for permission to seek access to the line.

ACCESS—A station has a connection.

Stations that already have access take the first free slot available to them after the appropriate START. Allocation of freed up slots on a reasonably equitable basis would proceed as follows. Stations would be aware of say n slots becoming free from the position of the END bit on the return line. Note, no END is issued if the class is full. Stations in the DEFER state would be permitted to compete for the n empty slots at the end of the subcycle. This will favor stations close to the head end. However, a large degree of fairness is achieved by permitting stations to switch from the WAIT to DEFER state only when two consecutive ENDS are encountered for that subcycle. This will only occur when all traffic currently in DEFER state has been granted access. At this point, traffic in the WAIT state would switch to DEFER and then vie for empty slots as they become available. This strategy is related to the snapshot algorithm.[16]

(*iv*) Continuation—In contrast to blocking, continuation requires that traffic not able to access the link in the previous cycle be served before any new traffic is accommodated. This may be achieved in the following manner. Assume that the class type is non-blocking. If the head station should issue a new START before all traffic of the class has been served, the end station will not detect the end of the cycle and hence will not issue END = 1. The absence of an END = 1 would indicate to the head station that the p registers of the stations in that class should not be reset on the next cycle (i.e., the stations would not switch from WAIT to DEFER).[15] Thus, in the following cycle, remaining traffic would be served. For centralized control, the START for this traffic type could contain an additional bit to indicate whether the previous cycle is being continued for deferring traffic or a fresh cycle is being started for new traffic. For distributed control, each station could keep track of the sequence of STARTs and ENDs.

It is important that the control strategy be adaptive to changing traffic conditions. We expect that the traffic mix will change relatively slowly—over a period of seconds rather than ms. Thus, it would be feasible to have the adaptation achieved by a server process.

The control algorithm could be implemented as completely distributed, completely centralized, or somewhere in between. Economics and reliability will dictate, to a large extent, where the control should be placed. Nevertheless, a hybrid strategy would seem more in the spirit of the current design. For example, selection is probably best achieved by having the head station transmit the appropriate START code (centralized, but perhaps assumable), whereas traffic assignment and continuation is probably best achieved by having each station read and operate on the END field (distributed).

VIII. TOPOLOGY

8.1 Introduction

A population of stations may be connected together by either a single link (Fig. 9a) or by several interconnected links (Fig. 9b). The best topology will depend upon physical distribution, traffic patterns, and the particular performance measures that one seeks to optimize. We will not consider the general problem, but restrict ourselves to the linear interconnection of links forming closed loops. Fasnets may be connected as shown in Fig. 10. Packets in Fasnet 1 destined for Fasnet 2 are addressed to station S_N. Station S_N transmits the packets to station S_1 which puts them on Fasnet 2. Similarly, for packets from Fasnet 2 destined for Fasnet 1. To provide reliability against single-station failures, interconnection stations would be provided in pairs. Thus, a similar connection would be provided between S_{N-1} and S_2. A detailed procedure may be specified whereby control of the interconnection passes from the $S_N - S_1$ connection to the $S_{N-1} - S_2$ connection in case of failure of the former. In principle, the secondary connection monitors both Fasnets and assumes the interconnection function after a suitable time-out period in event of failure. Provision can also be made for the primary connection to periodically check that the secondary connection is operational.

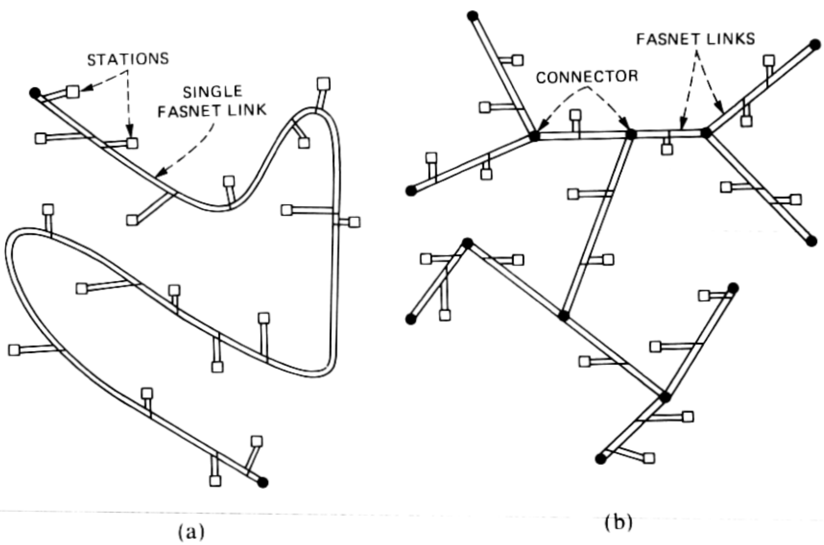

Fig. 9—(a) A cluster of stations being served by a single Fasnet link. (b) The same station population being served by several interconnected links.

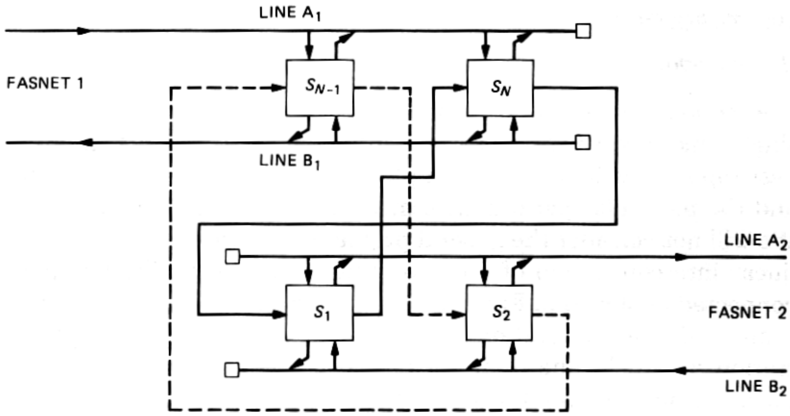

Fig. 10—Structure of a connector used to interconnect two Fasnet links.

Interconnection of Fasnets as shown in Fig. 10 permits traffic to pass from one link to another with a minimum of delay. Since the connection is to the first station on the link, the incoming packet can utilize the next occurring slot. Because of differences in frame timing between two links, it may be necessary to buffer a maximum of one complete packet; this amount of buffering is normally provided in a station interface. In general, interconnection of more than two Fasnets will require larger buffers to be employed to handle the condition where traffic arrives simultaneously for one Fasnet from connecting Fasnets.

8.2 Traffic localization

If stations on a Fasnet have traffic destined only for stations in the immediate vicinity, then total utilization can be significantly improved by dividing the single link into separate links that are connected. Only traffic that has not reached its destination link is allowed to cross the connector.

Consider two Fasnet links connected to form a ring with an inner and outer loop as shown in Fig. 11. The two connectors transfer traffic from one link to the other.

Assume that the ring has a length of unity and that the first link has a length l where $l \leq ½$. For a given packet, let d be the distance of the destination station from the source station. This distance is measured along the ring with the anticlockwise direction as positive, the clockwise direction as negative, and the source station as the origin. Assume that d is a random variable with a uniform distribution over $[-½, ½]$. When $d \epsilon [-½, 0]$, the source station accesses the outer loop; when $d \epsilon [0, ½]$, the source station accesses the inner loop. Thus, the source station selects the shortest distance to the destination along the ring.

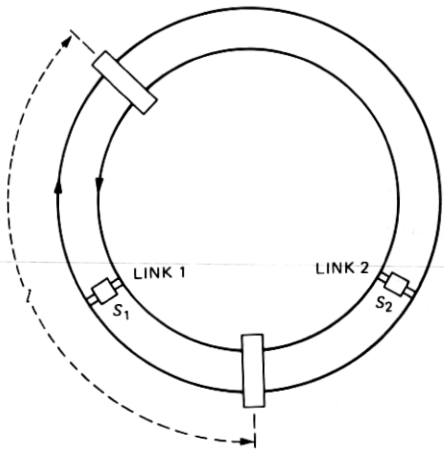

Fig. 11—Two Fasnet links interconnected to form a link using two connectors.

Depending on l and d, the traffic to any destination may have to traverse zero, one, or two connectors before it can be removed.

We first wish to determine the probabilities of the above events when the stations are distributed uniformly on the ring. For station S_j selected at random on link $j\,(j = 1, 2)$, let

p_j = probability that the destination is on the inner loop, and the same link, and the traffic traverses zero connectors,

q_j = probability that the destination is on the inner loop, but the other link, and the traffic traverses one connector,

r_j = probability that the destination is on the inner loop, and the same link, but the traffic traverses two connectors ($r_j \neq 0$ only if the length of the link is >½ and the shortest path between two stations at the ends of the link is through the other link).

It can easily be shown that

$$
\begin{aligned}
p_1 &= \frac{l}{2} & p_2 &= \frac{1}{2}\left(\frac{3}{4} - l\right)\frac{1}{1-l} \\
q_1 &= \frac{1-l}{2} & q_2 &= \frac{1}{2}\,l \\
r_1 &= 0 & r_2 &= \frac{1}{2}\,\frac{\left(\frac{1}{2} - l\right)^2}{1-l}. \\
&\left(\text{as } l < \frac{1}{2}\right)
\end{aligned}
\qquad (3)
$$

Note that $p_j + q_j + r_j = ½$, $(j = 1, 2)$ as any station accesses the inner loop with probability ½. Now suppose that there are N_1 active stations on link l and N_2 active stations on link 2. If each active station requires unit capacity, then the average traffic $T_i (i = 1, 2)$ in units of capacity on the inner loop in links 1 and 2 is given by

$$T_1 = N_1(p_1 + q_1 + r_1) + N_2(q_2 + r_2) + N_1 r_1$$
$$T_2 = N_2(p_2 + q_2 + r_2) + N_1(q_1 + r_1) + N_2 r_2 \tag{4}$$

because the traffic on a line in any link is the sum of three components: (*i*) all the traffic generated in that link, (*ii*) that fraction of the traffic generated in the other link that traverses this link, and (*iii*) that fraction of the traffic generated in this link that transverses the other link and then returns to the first link. If N_1 and N_2 are very large, then by the law of large numbers we have

$$\text{traffic on link 1} = T_1 \le C$$
$$\text{traffic on link 2} = T_2 \le C, \tag{5}$$

where C is the line capacity.

It can be shown that $N_1 + N_2$ is maximum when $l = ½$ and

$$N_1 + N_2 = \frac{8}{3} C.$$

If no connectors were used, we have a single Fasnet link (it can no longer be a closed ring) for which

$$N_1 + N_2 = 2C.$$

Hence the gain $G = (8C/3)/2C = 4/3$. Thus, we are able to obtain a 33 percent increase in the effective network capacity even for uniformly distributed traffic by having two diametrically located connectors.*

We now extend the above analysis to the case of K connectors C_1, C_2 $\cdots$ C_K, which are located symmetrically around the ring as shown in Fig. 12.

It can be shown by similar means that

$$G = \frac{4}{1 + \frac{4}{K}}. \tag{6}$$

This is plotted in Fig. 13.

We then extend the same analysis to the case with K symmetric connectors C_1, C_2 $\cdots$ C_K, but with an arbitrary traffic distribution symmetric about the source station and extending from −½ to +½

* While this analysis does not pertain to a specific access protocol, the effective gain can be closely realized by the Fasnet protocol and the connector structure of Fig. 10.

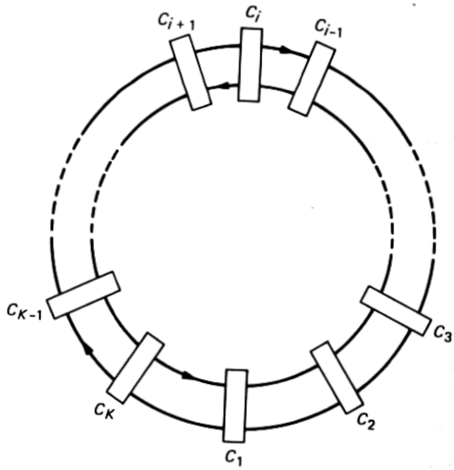

Fig. 12—A Fasnet link with multiple connectors.

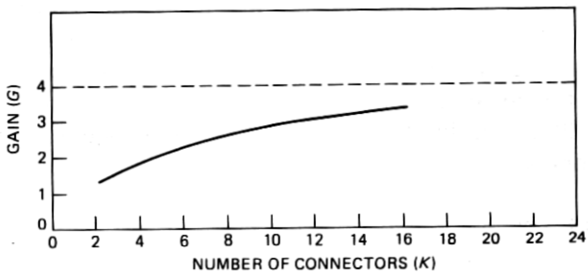

Fig. 13—Plot of the gain in traffic handling capacity of interconnected links relative to a single link as a function of K, the number of connectors.

along the ring. This preserves the shortest path routing, as well as sharing the load equally between the two loops.

It can be shown by similar means that for large K

$$G \approx \frac{K}{1 + KD} = \frac{1}{\frac{1}{K} + D}, \tag{7}$$

where D is the expected value of $|d|$, the absolute value of the source-destination distance. The approximation becomes exact if the distribution is uniform or if $K \rightarrow \infty$. Note that for the uniform distribution, $D = 1/4$ and

$$G = \frac{1}{\frac{1}{K} + \frac{1}{4}} = \frac{4}{1 + 4/K}$$

as before. For a given traffic distribution, (7) is a good design formula for how capacity costs can be reduced at the cost of extra connectors. However, this trade-off is useful only if accurate estimates of capacity and connector costs are available.

The above analysis highlights certain interesting features. For the uniform traffic distribution, it is seen that the gain, G, does not increase uniformly with K. This is fairly intuitive. Since the traffic is uniform, an extra connector, when K is large, removes the traffic over only a short link and results in only a marginal increase in the gain. On the other hand, as the traffic distribution becomes more localized (i.e., $D \rightarrow 0)G$ increases uniformly with K. This is again fairly obvious, as with a high degree of localization the traffic on each link is almost independent of the traffic on the other links.

We have considered here some configurations of interconnected links. There are interesting graph theoretic questions relating to reliability. For example, given a graph like Fig. 9b, what is the minimum number of additional links and their position so that full connectivity is still maintained if any link is cut at a single point? Development of realistic models of the physical traffic and cost structures of local environments still remains. There is a paucity of statistics (except for Ref. 2) on the local network parameters. Future operational local networks will hopefully furnish statistics on which to build more accurate models.

IX. CONCLUSION

The physical configuration of Fasnet consists of two communication lines passing each station. One line carries traffic in one direction, while the other line carries traffic in the opposite direction. Thus, this configuration carries twice the traffic of a previous system in which the two lines were connected at one end so that traffic was written on the outbound line and read from the inbound line. Each station makes two connections to each line, a nondirectional read tap and a directional write tap. Reliability of the physical medium is high because it contains no active electronics. The access protocol is partly centralized in that bit synchronization, framing, and start-of-cycle are provided by the end stations; however, these functions would be assumable by any station upon failure of an end station.

The access protocol is as follows: Upon reading a start-of-cycle, a station may transmit a prespecified number of packets in the first available empty slots. When all stations have transmitted their packets, a signal is sent on the return line to inform the head station to start a new cycle. The efficiency of Fasnet increases as the length of a cycle increases; cycle length depends upon the length of a packet, the number of active stations, and the number of packets, $p_{\max}$, that each station is permitted to send in a cycle. By adaptively changing p_{max}, efficiency can be maintained at a high level even for a small number of active stations. A number of techniques for further improving efficiency are suggested. A trade-off is necessary between increasing the complexity of the protocol, on one hand, and the resulting small improvements in efficiency on the other.

Bit synchronization of the stations is achieved through adding an out-of-band pilot tone, while framing is achieved through a periodically inserted code word. A three-level bipolar line code is preferred.

The potential of Fasnet for operation at high transmission rates makes it attractive as a conduit for the various types of traffic that may flow in a business environment. Mechanisms have been proposed to implement blocking, delaying, and request-for-service operations that are needed if mixed traffic is to be handled efficiently within a single medium. These operations can be implemented centrally or they can be distributed. The low-level access operations are best distributed while the more complex operations are best centralized.

Fasnets may be interconnected to increase the load that may be carried or to improve reliability. Investigation has been restricted to the connection of Fasnets to form a ring. As the number of segments in the ring increases, the throughput first increases rapidly (assuming uniformly distributed traffic). After about five segments, the increase is very small. Exploration of other topologies presents a challenge.

X. ACKNOWLEDGMENT

We thank B. Gopinath and N. F. Maxemchuk of Bell Laboratories and Z. L. Budrikis of the University of Western Australia for helpful discussions during various stages of this work.

REFERENCES

1. R. M. Metcalfe and D. R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," Commun. of the ACM *19,* 7 (July 1976), pp. 395–404.
2. J. F. Shoch and J. A. Hupp, "Measured Performance of an Ethernet Local Network," Presented at Local Area Communications Network Symposium, Boston (May 1979), pp. 113–34.
3. The Ethernet, A Local Area Network, Data Link Layer and Physical Layer Specifications, DEC, INTEL, XEROX, Version 1.0 (September 30, 1980).
4. F. A. Tobagi, "Multiaccess Protocols in Packet Communication Systems," IEEE Trans. Commun. *COM-28,* 4 (April 1980), pp. 468–88.
5. J. O. Limb, "High Speed Operation of Broadcast Local Networks," Proc. ICC, June 1982.
6. M. E. Ulug, G. M. White, and W. L. Adams, "Bidirectional Token Flow System, Proc. Seventh Data Commun. Symp. (October 1981), pp. 149–55.
7. M. V. Wilkes and D. J. Wheeler, "The Cambridge Digital Communication Ring," Proc. Local Area Commun. Network Symp. (May 1979), pp. 47–61.
8. C. S. Skrzypczak and W. E. Falconer, "Bell System Planning of ISDN," Proc. ICC, June 1981, pp. 19.6.1–6.
9. N. F. Maxemchuk and H. A. Wilder, "Virtual Editing: I. The Concept," Office Information Systems Workshop, St. Maximim, France, October 1981.

10. W. Pferd, L. A. Peralta, and F. X. Prendergast, "Inactive Graphics Teleconferencing," Computer *12* (November 1979), pp. 62–71.
11. W. A. Rheinfelder, *CATV System Engineering*, Third Ed., Blue Ridge Summit, Pa.: TAB Books, 1972.
12. J. O. Limb, "Fasnet: Proposal for a High Speed Local Network," Office Information Systems Workshop, St. Maximim, France, October 1981.
13. IEEE 802 Standards Committee, Interim Report.
14. F. D. Waldhauer, "A 2-Level, 274 Mb/s Regenerative Repeater for T4M," Proc. of the ICC 1975, *3*, pp. 48-13–17.
15. L. Frata, F. Borgonovo, and F. A. Tobagi, "The Express-Net: A Local Area Communication Network Integrating Voice and Data," in *Performance of Data Communication Systems*, G. Pujolle, Ed., Amsterdam: North Holland, 1981, pp. 77–88.
16. R. C. Chen, "Bus Communication Systems," Report No. PB 235-897, Carnegie-Mellon University, January 1974.